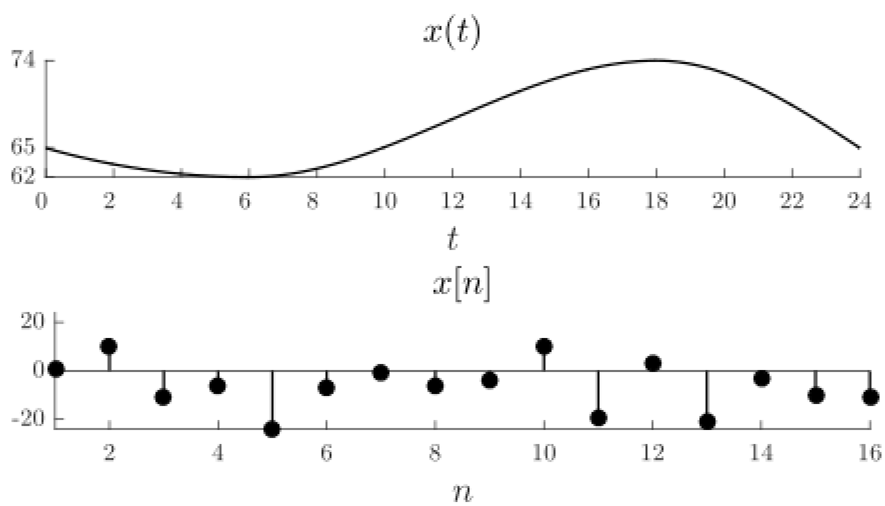
What are Signals?

A **signal** is a mathematical entity expressed as a function of some independent variable. As dry and abstract as that definition sounds, signals and the systems that modify them are actually interesting, important, and--believe it or not--very relevant to your life.

Signals are all around us. Think of any real-world phenomenon that can be mathematically quantified, and that also varies with respect to another variable, such as time. Go ahead, take a moment and think of one, what is something around you that can be expressed as a quantity that may change over time? What you have just thought of is a signal. The ambient temperature in your room over the course of a day is a signal. Your favorite sports team's margin of victory (or defeat) over the course of a season is also a signal.

The signal $x(t)$ represents the temperature in a room over the course of 24 hours. The signal $x[n]$ represents a team's margin of victory over the course of a sixteen game season (in this case, the disastrous 2015 season of the Dallas Cowboys).



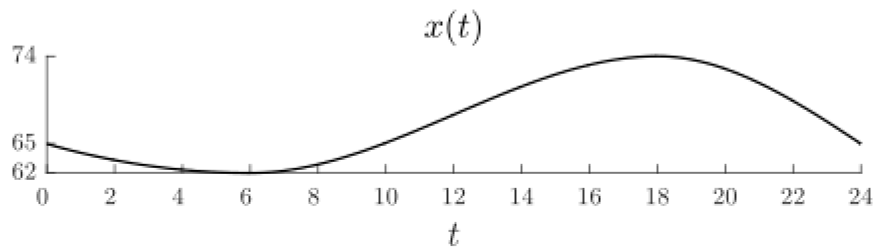
Those two examples are fairly trivial (though both may influence how you feel!). Other signals might affect you in more significant ways. The daily closing value of the Dow Jones average might say something about your savings, the electrical potential of your heart over time--also known as an electrocardiogram--might say something about your health, and the voltage

a radio wave induces on the antenna in your phone might (literally) say something directly to you.

Types of Signals

Signals are as varied and unique as the realities they express. That said, they can also be classified in a number of conventional ways. One of the most fundamental ways to categorize a signal is by the nature of its independent variable. Suppose, for instance, the independent variable is time. A signal may be expressed in terms of a continuum of time values, such as with the example of the temperature in a room over the course of a day. At any given time instant, there is a value for the signal: the temperature of the room at that moment in time. We can represent the time variable as a real number t, and the temperature at time t as $x(t)$. Signals with these types of independent variables are known as **continuous-time signals**. It will be our convention to refer to these signals using functions whose arguments are in parentheses--e.g., $x(t)$ or $y(t)$--and plot these signals with line plots (see [link]).

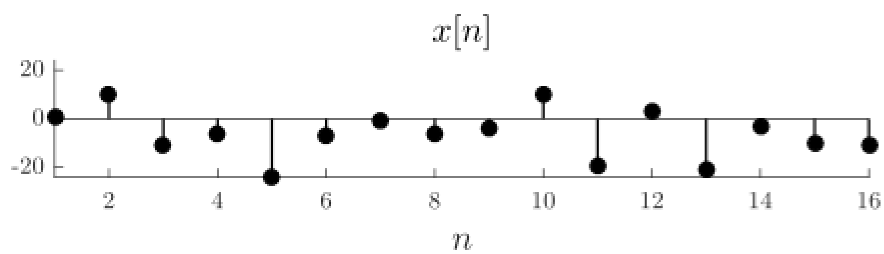
The temperature in a room over the course of 24 hours is a continuous-time signal. Note that the signal $x(t)$ is referred to with parentheses around the argument, because t runs continuously from $t = 0$ (representing midnight) to $t = 24$ (representing the following midnight).



Other signals do not have values for any arbitrary value of time. Consider the example of your sports team's margin of victory, or the daily closing price of a financial stock. These signals are defined only for certain discrete

values of time. The victory margin is only defined for particular games (in football, for example, as week number 1, or 2, or 3, etc.), and the daily closing price is only defined for weekdays. Outside of those particular units of time, the signals are not defined; it makes no sense to ask what the victory margin or closing price is at, say 12:01 p.m. as compared to 12:02 p.m. The independent variable for each of these examples can be expressed as an integer n, and the value of the signal at that time n is $x[n]$. Signals with integer-valued independent variables are known as **discrete-time signals**. As you may have already noticed, it will be our convention to use brackets to contain the arguments of these signals' functions--e.g. $x[n]$ or $y[n]$--and plot the signals with stem plots ([link]).

A sports team's margin of victory over the course of a season is a discrete-time signal. The time variable runs from 1 to 16, but not continuously, for there is no such thing as a final score for game number 2.397! Note that the signal $x[n]$ uses brackets around the argument because the time variable takes only discrete, integer values.

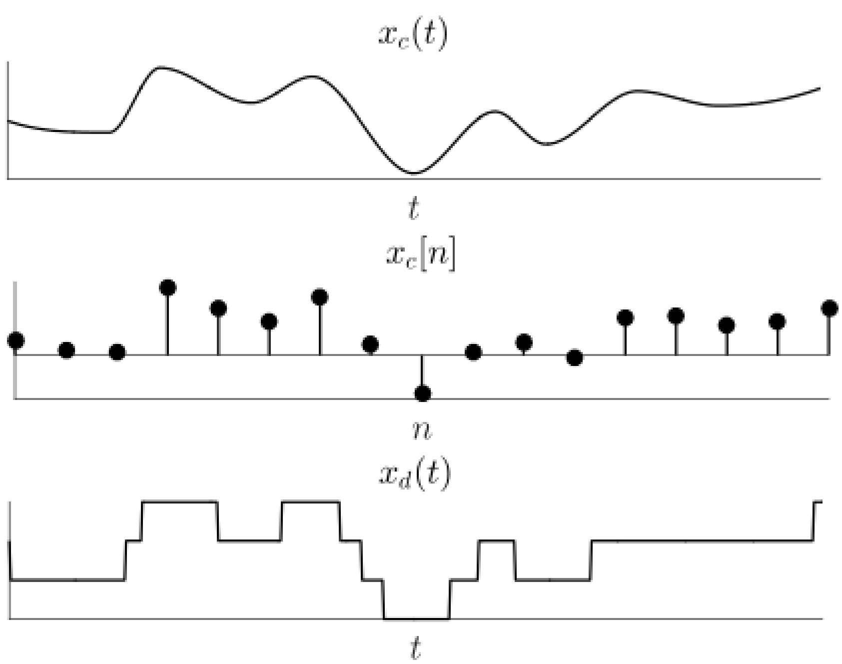


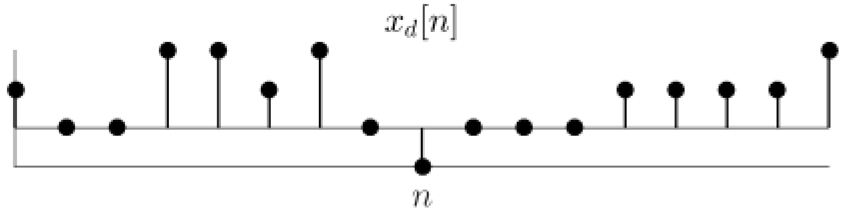
This general division of signals according to their independent variables can also be extended to their dependent variables, as well. The value a signal has at a particular time (be it t or n) could, like the time variable itself, be either continuous-valued or discrete-valued. A football team can only score an integer number of points, whereas the ambient temperature can vary at infinitesimally small increments (quantum mechanics aside). If a signal has both a continuous-valued independent and dependent variable (such as the temperature in a room at time t), it is known as an **analog** signal; most "real world" signals are analog. If both variables are discrete-valued, the signal is known as a **digital** signal; *any* signal stored in a computer is digital. There

are no commonly used terms for continuous-valued discrete-time signals, nor for discrete-valued continuous-time signals, even though you can probably think of real world examples of each. [link] shows plots of all four of these signal types.

For our purposes, we will always assume (unless otherwise stated) that both the continuous-time and discrete-time signals we deal with are continuous-valued. While we will obviously work with signals quite a bit on our computers--and hence, these signals are actually discrete-valued--the abundance of computer memory allows us to treat them as being essentially continuous-valued.

The possible values a signal may take at a particular time instance could reside either on a continuum or may be a discrete set of values. $x_c(t)$ and $x_c[n]$ are examples of continuous-valued signals, while $x_d(t)$ and $x_d[n]$ are discrete-valued. If a signal is continuous-valued and continuous-time, such as $x_c(t)$, it is called an analog signal. Discrete-valued discrete-time signals, such as $x_d[n]$, are called digital signals.




$x_d[n]$
n

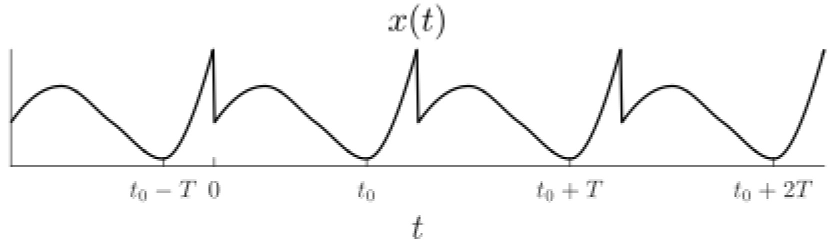
Properties of Continuous-Time Signals

As we have already seen, the most basic way of classifying signals is according to their independent variables, whether they are continuous or discrete. But that is not the only classification of interest. We will now consider other possible properties, focusing on continuous-time signals.

Periodicity

When dealing with a signal $x(t)$, one of the first things we would like to know about it is whether or not it repeats itself. That might not seem that important now, but this property will be called upon quite a bit as we study and analyze signals. We say a signal is **periodic** if it repeats at a constant rate. To put it mathematically, a signal is defined to be periodic if there exists some positive real number T such that $x(t+T) = x(t)$ for all t. The smallest T for which that identity holds is known as the **period**. If no such T exists, the signal is said to be **aperiodic**.

The signal $x(t)$ is periodic with period T.



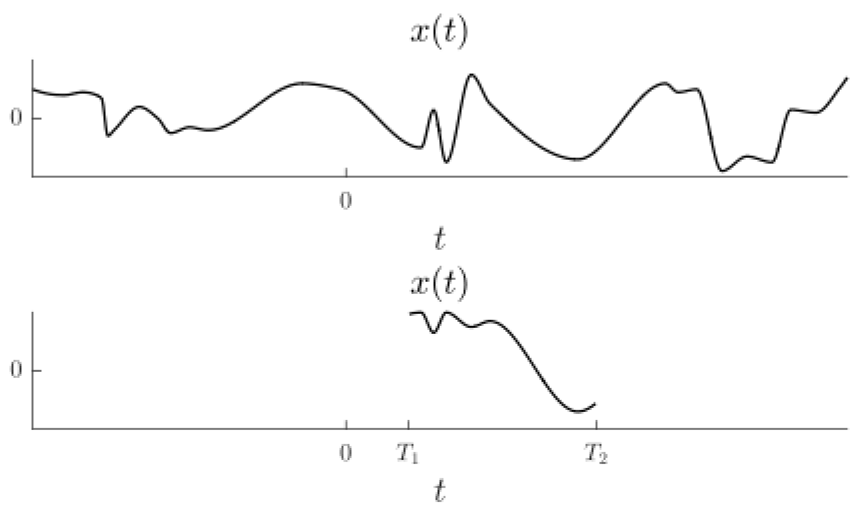
In [link], the signal $x(t)$ is periodic. Note how $x(t_0 + T) = x(t_0)$. Visual inspection confirms this is the case not only for t_0, but for any t. Because we have that $x(t+T) = x(t)$ for all t, it follows that $x(t+2T) = x(t)$, because $x(t+2T) = x(t+T)$. We can extend that to say that $x(t+mT) = x(t)$ for any integer m, which is simply another way of explaining that periodic signals repeat over and over.

Length

It is characteristic of a continuous-time signal $x(t)$ that its independent variable t is continuous, meaning that it is a real number (as opposed to the integer-valued discrete-time independent variable n for $x[n]$). But even among signals with this continuous type of independent variable, we may make a further distinction: is the signal defined for all possible values of the independent variable, or only for a limited section of possible values? If $x(t)$ is defined for all t on the real number line, then $x(t)$ is said to be **infinite-length**. If a signal is only defined for a particular interval of the independent variable's number line, then it is said to be **finite-length**.

Now, if you were to plot an infinite-length signal for all of its time values, your plot would extend infinitely far to the left and right! So in practice we obviously cannot plot the entirety of infinite-length signals, but rather only a portion of them (with the assumption they continue beyond our space-limited plots).

The first signal $x(t)$ is of infinite-length, even though obviously only a portion of it is shown. The second signal $x(t)$ is finite-length; note how it is only defined for the interval $T_1 \leq t \leq T_2$.

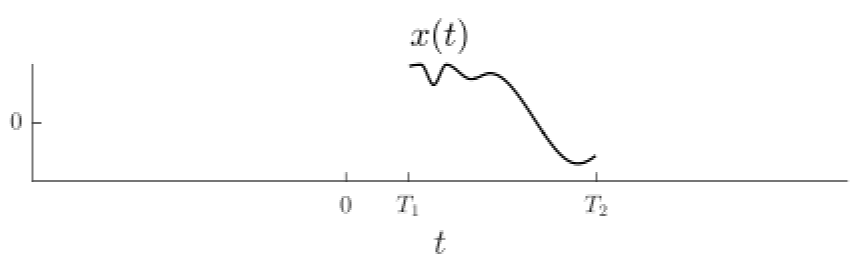


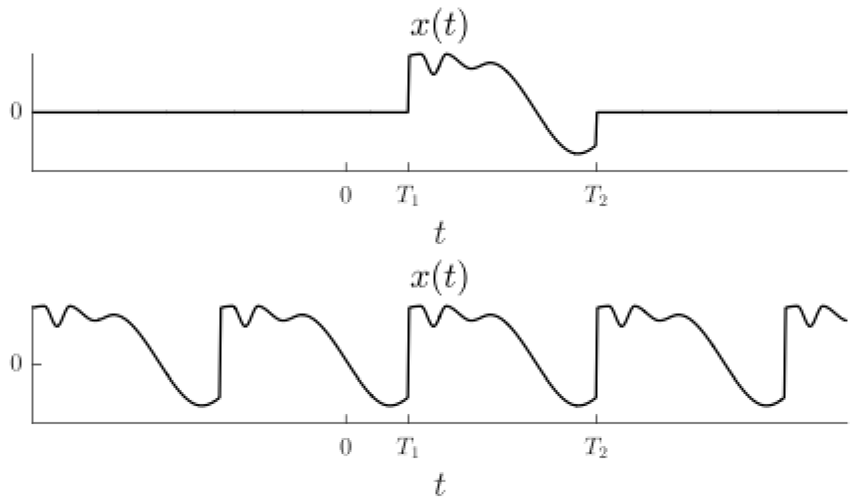
Since the a finite-length signal has no value outside a defined interval of t, say between T_1 and T_2, it might be tempting to think the signal actually has a value of 0 outside of those t--after all, "no value" and "0 value" are semantically similar. But this is not the case; outside the defined set of t, the signal is totally undefined; it is not 0 or anything else. Thus especially if we are given an explicit expression for a finite-length signal, we also need to know the values of the independent variable for which it is defined. So, $x(t) = 2t$ is an infinite-length signal, but there may be a finite-length signal that also equals $2t$ when it is defined, e.g. for this finite-length signal:

$$x(t) = \begin{cases} 2t & 1 \leq t \leq 4 \\ \text{undefined} & \text{else} \end{cases}$$

There may be times when we would like to convert a finite-length signal to an infinite-length one. There are many ways we could do this, but the following two are the most common. We could assign a value of 0 to all of the time-values for which the signal was previously undefined, a process known as **zero-padding**. Now you see why you should not have assumed that a finite-length signal is not 0 outside its defined interval; that was actually a strategy to make the signal infinite-length! The other option would be to **periodize** the signal, meaning to repeat it again and again before and after its defined interval. [link] shows plots of each of these ways to convert a finite-length signal to be of infinite-length.

Here we see two ways to convert a finite-length signal into an infinite-length one. We can either define it to be zero outside the originally undefined set of independent variable values, or we can periodize it by repeating the originally defined values over and over.



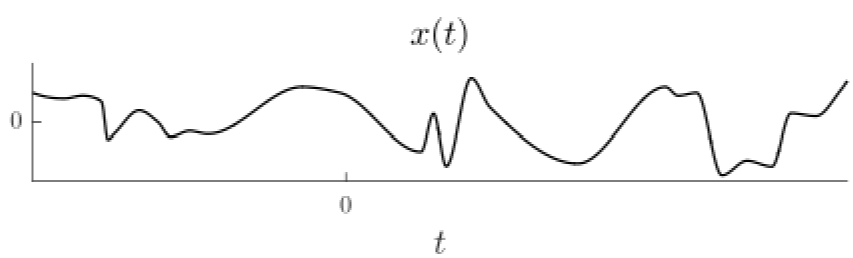


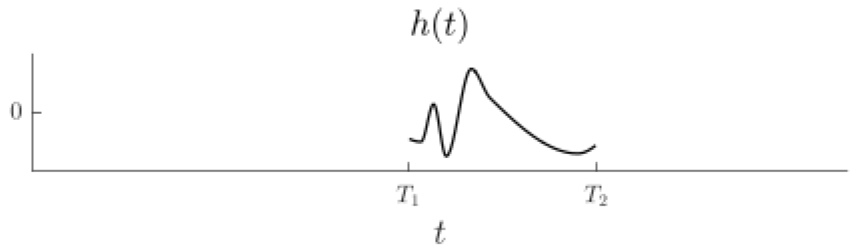
As we consider that second option--making a finite-length signal to be infinite-length by periodizing it--we can see that finite-length signals and periodic infinite-length signals are essentially equivalent. An infinite-length periodic signal can be understood to be simply be a periodized version of a finite-length signal corresponding to one period of the periodic signal.

It is also possible to take an infinite length signal and create a finite-length signal from it. Again, there may be many ways to do this, but the most common is to simply cut out a portion of it, e.g.:

$$h(t) = \begin{cases} x(t) & T_1 \leq t \leq T_2 \\ \text{undefined} & \text{else} \end{cases}$$

Here an infinite-length signal $x(t)$ (of which only a portion is shown) is converted into a finite-length signal $h(t)$ by extracting a portion of it.





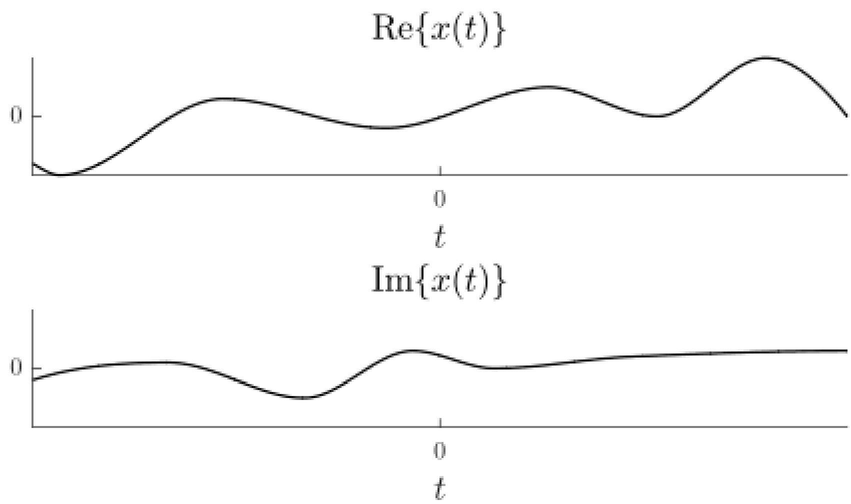
Real and Complex Signals

Up to this point all of the signals we have seen were **real-valued**, meaning that their value at any particular time was a real number. As we continue to study continuous-time signals, we will soon come across **complex-valued** signals, those that have a complex value (i.e., a real component and an imaginary component) for each time value. Now, the idea of dealing with complex numbers may discourage you. This is understandable; after all, imaginary numbers don't really exist in real life (you can't buy $\sqrt{-1}$ apples at the store). The reason we deal with complex numbers is not because they are actual quantities, but because they are useful. For example, when tossing a tennis ball before a serve, it is helpful to pretend you are holding a glass of water in the palm of your hand, or that you are pushing the ball through a vertical pipe.[footnote] The reason you do this is not so that you become good at holding water glasses or pushing balls through pipes, but because imagining the toss in that way results in a straighter and more consistent toss. In the same way, it can be helpful at times for us to imagine that real-world signals correspond to the real part of some underlying complex-valued signals; the reason why is because complex-valued signals are often easier to work with than real signals.
See https://www.usta.com/Improve-Your-Game/Player-to-Player/player_to_player_consistent_service_toss/

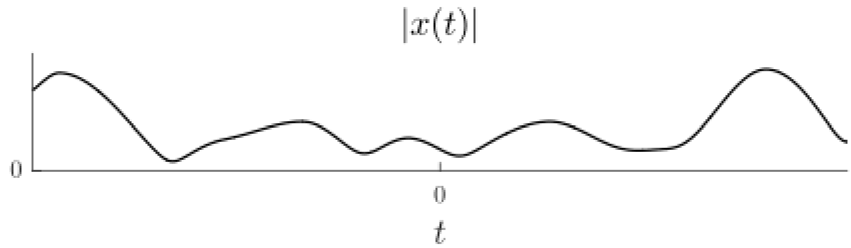
Just as with complex numbers, a complex signal $x(t)$ has a real part $\mathrm{Re}\{x(t)\}$ and an imaginary part $\mathrm{Im}\{x(t)\}$. Putting those together, we have that $x(t) = \mathrm{Re}\{x(t)\} + j\mathrm{Im}\{x(t)\}$, where j represents the imaginary number $\sqrt{-1}$. Note that in other contexts it is represented as i, but in engineering i instead represents electrical current intensity (usually just called "current").

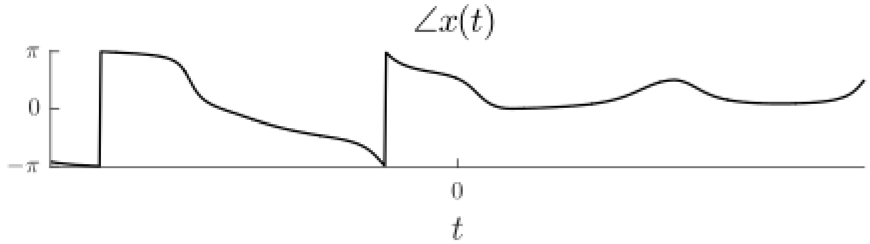
Complex-valued signals can either be plotted in terms of their real and imaginary parts ([link]) or in terms of their magnitude and phase ([link]). With the latter, note how the magnitude above is always greater than or equal to 0, and the phase is always between $-\pi$ and π. The phase may also, equivalently, be expressed between 0 and 2π.

A single complex-valued signal must be displayed in two plots; here the real and imaginary parts are plotted.



Complex-valued signals can also be plotted in terms of their phase and magnitude at each time point t.





$\angle x(t)$
π
0
$-\pi$
0
t

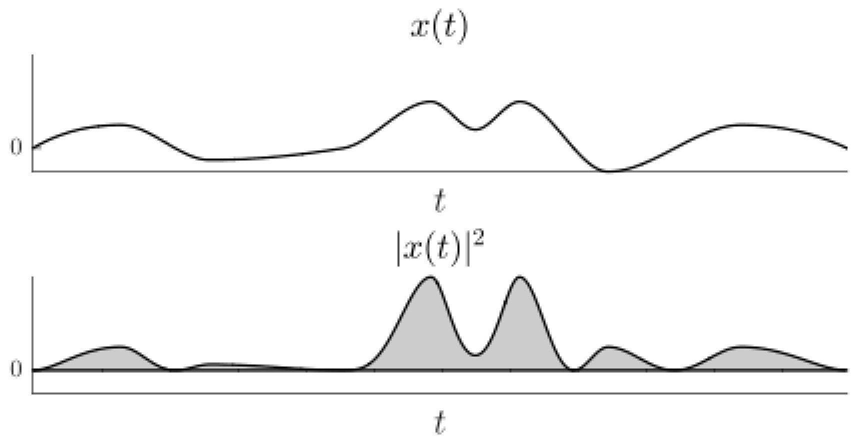
Norms of Continuous-Time Signals

Among the many things we might observe or wish to know about a given continuous-time signal, another property of interest is the signal's strength: how *big* is the signal? As with physical strength, there are a number of different ways we can measure a signal's strength. One way is to calculate a signal's **energy**. For a signal $x(t)$, the energy is defined as:

$$E_x = \int_{-\infty}^{\infty} |x(t)|^2 dt,$$

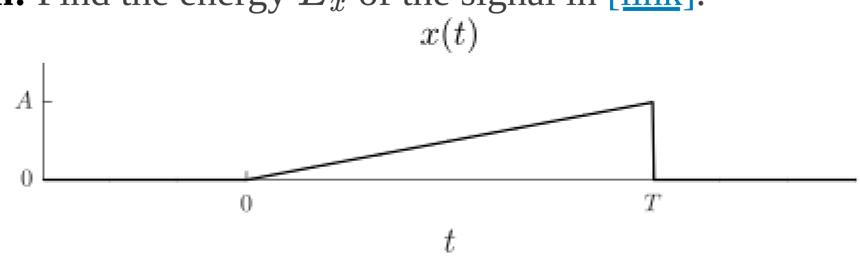
which is simply the area under the curve $|x(t)|^2$ ([link]).

The energy of a signal $x(t)$ is defined as the area under the curve $x(t)$.



Exercise:

Problem: Find the energy E_x of the signal in [link].



Solution:

$$
\begin{aligned}
E_x &= \int_{-\infty}^{\infty} |x(t)|^2 dt \\
&= \int_0^T |\frac{A}{T}t|^2 dt \\
&= \int_0^T \frac{A^2}{T^2}t^2 dt \\
&= \left[\frac{A^2}{3T^2}t^3\right]_0^T \\
&= \frac{A^2T^3}{3T^2} \\
&= \frac{1}{3}A^2T
\end{aligned}
$$

Signal Norms

For our purposes, we will usually refer to a signal's strength with reference to its **norm**, which is a kind of generalized measurement of the signal's energy. The L_p norm of a signal $x(t)$ is defined as:

$$\| \, x(t) \, \|_p = \left(\int_{-\infty}^{\infty} |x(t)|^p dt \right)^{1/p}, \; 1 \leq p < \infty.$$

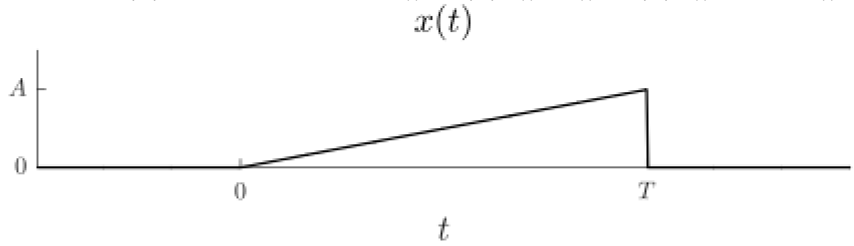
We will most often use values of 1 and 2 for L_p norms, as well as ∞, which is defined as:

$$\| \, x(t) \, \|_\infty = \text{ess sup } |x(t)|$$

"Ess sup" means the essential supremem of $x(t)$. That has a fairly technical mathematical definition, but for our purposes it is simply the maximum value of $|x(t)|$.

Exercise:

Problem: For $x(t)$ in [link], find $\| x(t) \|_1$, $\| x(t) \|_2$, and $\| x(t) \|_\infty$.



Solution:

$$\begin{aligned}
\| x(t) \|_1 &= \int_{-\infty}^{\infty} |x(t)| dt \\
&= \int_0^T |\frac{A}{T} t| dt \\
&= \int_0^T \frac{A}{T} t dt \\
&= \left[\frac{A}{2T} t^2 \right]_0^T \\
&= \frac{AT^2}{2T} \\
&= \frac{1}{2} AT
\end{aligned}$$

$$
\begin{aligned}
\| x(t) \|_2 &= \int_{-\infty}^{\infty} |x(t)|^2 dt \\
&= \int_0^T |\frac{A}{T} t|^2 dt \\
&= \int_0^T \frac{A^2}{T^2} t^2 dt \\
&= \left[\frac{A^2}{3T^2} t^3 \right]_0^T \\
&= \frac{A^2 T^3}{3T^2} \\
&= \frac{1}{3} A^2 T
\end{aligned}
$$

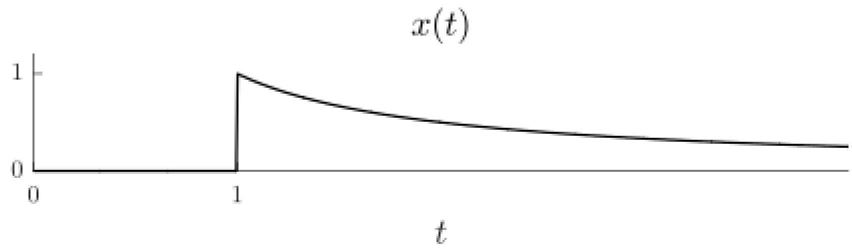
$$\| x(t) \|_\infty = A$$

L_p Signal Spaces

Having defined the L_p norms for signals, it should be evident that they do not necessarily always exist. It may be that they are unbounded for certain signals and/or values of p. For example, the following signal's L_1 norm does not exist, but its L_2 and L_∞ norms do exist:

$$x(t) = \begin{cases} \frac{1}{t} & t \geq 1 \\ 0 & t < 1 \end{cases}$$

For this signal $x(t)$, the L_1 norm is unbounded, while the L_2 and L_∞ norms do exist. Can you calculate them?



We can broadly categorize infinite-length signals depending on whether or not they have finite L_p norms. The L_p **signal space** for a particular value of p is the set of all signals that have a finite L_p norm:

$$L_p(\mathbb{R}) = \{x(t) : \| x(t) \|_p < \infty\}$$

(the $\mathbb{R}$ simply means we are talking about signals that have the real number line as their independent variable)

Signal Power

There may be times in which a signal of interest does not have a finite norm, such as with a periodic signal (because it repeats forever, its norm will be unbounded). We may wonder if there is a way to convey the strength of these signals, and for many such cases there is. We can examine what the energy (which, you recall is similar to the norm) of a signal is *per unit of time*. We will refer to this quantity as the signal's **power**, and it is defined over a certain time period T as:

$$P_x = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} |x(t)|^2 dt$$

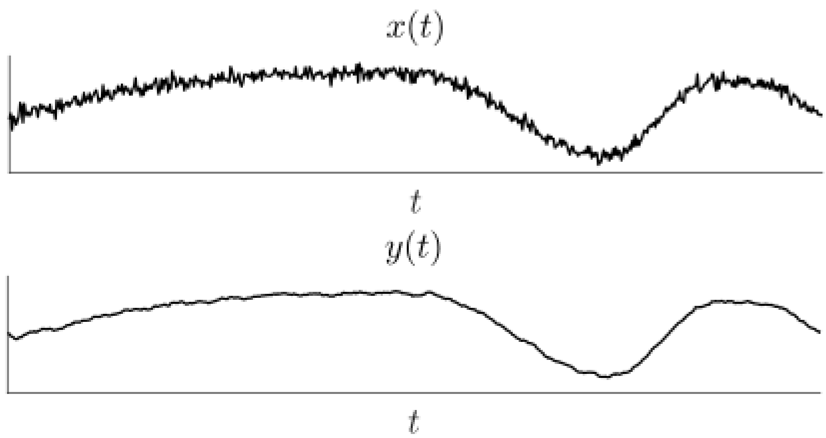
As $T \to \infty$, this value is known as the **average power** of the signal over all of its time. If we then take the square root of the average power, we have what is called the **root mean squared (RMS)** value of the signal:

$$RMS_x = \sqrt{\lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} |x(t)|^2 dt}$$

Continuous-Time Systems

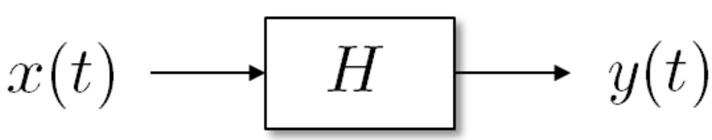
We have learned what signals are and the various ways they are typically classified. But we want to do more than simply observe and classify signals, we would like to *do* something with them. For example, we may want to take a noisy signal $x(t)$ and try to reduce the noise, calling the new signal we produce $y(t)$ ([link]).

A system modifies an input signal and produces a new signal as an output. A de-noising system can take a noisy signal such as $x(t)$ and produce a less noisy version of it, $y(t)$.



The entity that processes an input signal and produces an output signal is called a **system**. We will typically refer to the **input** signal as $x(t)$ and the **output** as $y(t)$, and sometimes give the system a name, such as H. When H operates on the input $x(t)$, we can refer to that operation as $H[x(t)]$. The input/system/output relationship can also be represented pictorially, as shown in [link].

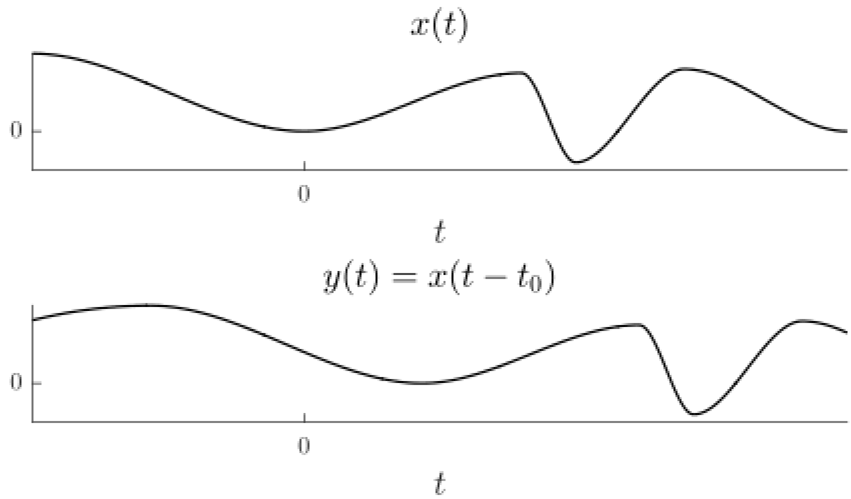
A system may be represented as a box having an input and and output.



There are all sorts of different kinds of systems, as many as the different ways we may wish to modify a signal! We have already seen that we may

want to de-noise it. Another common modification is to delay a signal. A system H may take an input $x(t)$ and produce output $y(t) = x(t - t_0)$ ([link]).

Here a system takes an input $x(t)$ and delays it by a time of t_0, producing an output $y(t) = x(t - t_0)$. In this case the value of t_0 is positive.



Or we may want to amplify a signal, or attenuate it, or mix it with another signal. Supposing the signal is a picture, we may want to de-blur it (or perform one of the many other image enhancement techniques offered by an image editing program). If the signal is a music recording, we may want to turn up the bass. For signals that are stock prices or global temperature measurements, we may want to create new signals that predict future values. For each of these and a myriad of other desired signal modifications, there is a system that takes the input signal, operates on it, and produces the desired output.

So there are many different kinds of systems, but just as with signals, there are several ways we can broadly categorize them by what they do. Four classifications are particularly important: whether or not the system is linear, time-invariant, causal, and/or stable.

Linearity

A continuous-time system H is **linear** if it obeys the following two properties for any signals $x_1(t)$ and $x_2(t)$ and constant value $\alpha \in \mathbb{C}$:

- Scaling: $H[\alpha x_1(t)] = \alpha H[x_1(t)]$
- Superposition: $H[x_1(t) + x_2(t)] = H[x_1(t)] + H[x_2(t)]$

The **scaling property** says that scaling the input a certain amount will result in the output be scaled the same amount; e.g., if an input produces a certain output, doubling the input and putting it back through the system will result in an output double the size of the original output. A system satisfies the **superposition property** if an input that is the sum of two signals produces an output the same as summing the two outputs of the signals when input individually. Note that in order to prove a system is linear, we must show that both of these properties hold for arbitrary signals and constants. To show a system is not linear, we only need to provide a single counter-example of the system not holding one of the properties.

Example:
Let's consider a system that adds a constant value of 1 to any input and see if it is linear.
$H[x(t)] = x(t) + 1$
We will first test the scaling property:
$$\begin{aligned} H[\alpha x(t)] &= (\alpha x(t)) + 1 \\ &= \alpha(x(t) + 1) + (1 - \alpha) \\ &= \alpha H[x(t)] + (1 - \alpha) \end{aligned}$$
So the system will only satisfy the scaling property (i.e., have $H[\alpha x(t)] = \alpha H[x(t)]$) for $\alpha = 1$ (which, of course, is not scaling the input at all!). If α is any other number then the property will fail. Thus the system is not linear.
To disprove linearity, we could have merely provided a single counterexample. Suppose $x(t) = 0$ and $\alpha = 2$. $H[x(t)] = 1$, yet $H[2x(t)] = 1 \neq 2H[x(t)]$.

Exercise:

Problem:

Work out for yourself whether or not the following systems are linear:

- $H[x(t)] = (x(t))^2$
- $H[x(t)] = x(t^2)$
- $H[x(t)] = 2x(t) + x(t-2)$
- $H[x(t)] = 2\frac{d}{dt}x(t)$

Solution:

- $H[x(t)] = (x(t))^2$ -- not linear
- $H[x(t)] = x(t^2)$ -- linear
- $H[x(t)] = 2x(t) + x(t-2)$ -- linear
- $H[x(t)] = 2\frac{d}{dt}x(t)$ -- linear

Time-invariance

The second significant property of a system is whether or not it is **time-invariant**. A system H is said to be time-invariant if, for an arbitrary time delay t_0, and an arbitrary input $x(t)$ whose output is denoted $y(t)$, $H[x(t-t_0)] = y(t-t_0)$. Time-invariance means that delaying an input (i.e., putting it into the system at a different time) will produce a corresponding delay in the output. Time-invariance is a property that we intuitively expect most any system to have--we generally want systems to operate the same way in the evening as in the morning, or on Tuesday as on Wednesday.

Example:
Let's look at a couple simple systems and try to determine whether or not they are time-invariant. The first one is $H[x(t)] = x(t-3)$. So

$y(t) = x(t-3)$. What we would like to know is if $H[x(t-t_0)] = y(t-t_0)$. It is helpful when examining time invariance to introduce a placeholder function $g(t) = x(t-t_0)$ for the intermediate steps.

$$\begin{aligned} H[x(t-t_0)] &= H[g(t)] \\ &= g(t-3) \\ &= x((t-3)-t_0) \\ &= x((t-t_0)-3) \\ &= y(t-t_0) \end{aligned}$$

So this system is time-invariant. Now let's look at another simple system, $H[x(t)] = x(t) - t$. So $y(t) = x(t) - t$ and we will want to show that $H[x(t-t_0)] = y(t-t_0)$. Again, we will use a placeholder $g(t) = x(t-t_0)$ to help keep all of the t's and t_0's clear:

$$\begin{aligned} H[x(t-t_0)] &= H[g(t)] \\ &= g(t) - t \\ &= x(t-t_0) - t \\ &= (x(t-t_0) - (t-t_0)) - t_0 \\ &= y(t-t_0) - t_0 \\ &\neq y(t-t_0) \end{aligned}$$

So this system is not time-invariant.

Exercise:

Problem:

Work out for yourself whether or not the following systems are time-invariant:

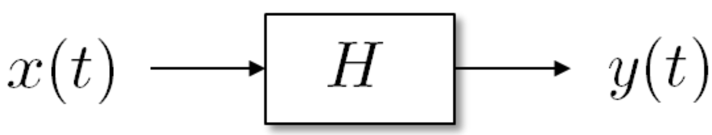
- $H[x(t)] = (x(t))^2$
- $H[x(t)] = x(t^2)$
- $H[x(t)] = 2x(t) + x(t-2)$
- $H[x(t)] = 2\frac{d}{dt}x(t)$

Solution:

- $H[x(t)] = (x(t))^2$ -- time-invariant
- $H[x(t)] = x(t^2)$ -- time-varying
- $H[x(t)] = 2x(t) + x(t-2)$ -- time-invariant
- $H[x(t)] = 2\frac{d}{dt}x(t)$ -- time-invariant

Causality

Recall that a system takes an input $x(t)$ and produces an output $y(t)$ ([link]). The very nature of that diagram suggests a temporal process: $x(t)$ goes in, and then $y(t)$ comes out. Or, in other words, $y(t)$ only depends on the present (and perhaps also the past) values of $x(t)$. And at first, it makes sense practically speaking as well, that a system cannot act in the present based upon future values of its input. But in theory, a system certainly *could* be defined to consider future values of $x(t)$. If a signal is not processed in real-time, then at a given "current" time the output could indeed took to "future" values of the input, e.g., $y(t) = x(t+1)$.



Because it is possible to define a system that looks into the future, we need a special definition for the systems that don't do that. A system is said to be **causal** if the output at any given time depends only on the present and/or past values of the input.

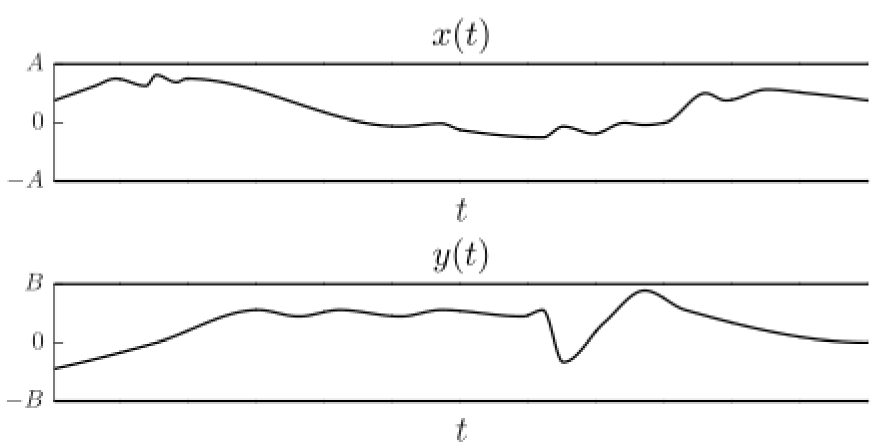
It's not too difficult to determine if a system is causal or not, provided its input/output relationship is known. All one needs to do is look at it and see if it ever takes future values of time into account. So, for example, the following system is causal because it never considers future values of the input:

$y(t) = 2x(t-1) + x(t)^2$
But now consider this system:
$y(t) = x(t^2)$
For all time values in which $|t|$ is greater than 1 , the output $y(t)$ is a future value of the input, e.g. $y(2) = x(4)$, so the system is not causal.

Stability

Imagine having a phone that always worked perfectly, but then its circuits would get fried if you sneezed. That would not be a good response! Or if an ECG blew up if a person's heart rate increased too quickly (also not good), or your radio shut off whenever a song from Kesha's first studio album came on (understandable). If a system has a reasonable input, we want the output not to be anything too unexpected. In other words, we would like systems to be **stable**.

There are various mathematical ways of certifying a system to be stable, but the one we will focus on is **bounded-input bounded-output (BIBO) stability**. As you might surmise, this describes a system that, so long as the input is bounded by some amplitude, will have an output also bounded by some (different) amplitude. Mathematically, a system H is said to be BIBO stable if, for any signal $x(t)$ such that $|x(t)| < A$, there exists some B such that its output $y(t)$ is also bounded, $|y(t)| < B$ ([link]).



BIBO stability is a guarantee for systems. If a system is BIBO stable, then no matter what sort of input it receives (so long as the magnitude is bounded), the output is going to be bounded.

Continuous-Time Impulse Response and Convolution

Recall that a continuous-time system is an entity that takes a continuous-time signal as an input and produces another continuous-time signal as an output. A task we will often face is determining what a system's output will be for a given input. Sometimes, this could be trivially easy. For example, if a system is defined as $H[x(t)] = x(t-1) + \frac{1}{2}x(t-2)$ and the given input is $x(t) = \sin(t)$, the output is simply $y(t) = \sin(t-1) + \frac{1}{2}\sin(t-2)$.

In many cases, finding the output may be a little more complicated. We might not be given an explicit input/output relationship, and instead only have physical measurements of the system (see "impulse response," below). Or the input/output relationship might be tricky to work with; systems built with capacitors and resistors will have an input/output relationship that is a differential equation, e.g.:

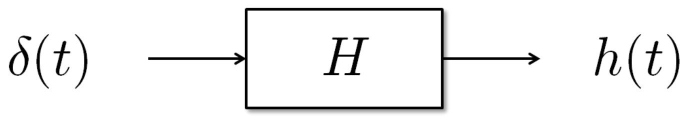
$$y(t) = b_0 x(t) + b_1 x'(t) + b_2 x''(t) - a_2 y''(t) - a_1 y'(t)$$

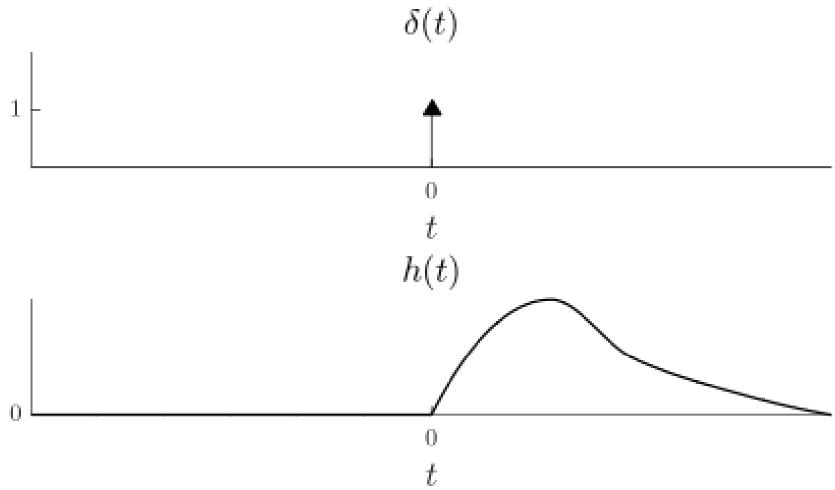
One way to find the output would be to solve the differential equation. That approach works fine, but the trouble is that for every new input, the differential equation will have to be re-solved to find the output, which might become somewhat tedious. For LTI systems, it turns out there is another way to find a system's output for a given input, one that only requires a straightforward computation of a single integral. Before we can reveal the nature of this special integral, there is one more important concept in continuous-time signals and systems we need to meet: the impulse response.

The Impulse Response of an LTI Continuous-time System

We will ultimately want to find a system's output $y(t)$ for a given $x(t)$, but before we do that, we first would like to know what the output is for a very specific input, the delta function $\delta(t)$. That particular output, $H[\delta(t)]$, is so special that it has its own name, the **impulse response**, and notation, $h(t)$.

For LTI systems, the system output when the input is a delta function has special significance.





The impulse response can be thought of in physical terms as using a delta function to somehow shock or probe a system in question, with the impulse response being the output. The reason the impulse response of LTI systems is so important is because it can be used to compute the output corresponding to any other input, and thus it contains all the information about the nature of the system. If we know the impulse response, we have complete knowledge of the system.

Convolution: Finding the output from the impulse response and input

If we are given a system's impulse response, we will be able to find the system's output for any given input. To see why, first recall the sifting property of the delta function, how it allows us to express an $x(t)$ in terms of delta functions:

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau$$

We will now see why a system being LTI is so helpful. Let's say we would like to put $x(t)$ through an LTI system H , so as to produce $y(t)$. We have that:

$$y(t) = H[x(t)] = H\left[\int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau\right]$$

Alright, so all we've done is put the integral form of $x(t)$ inside $H[\cdot]$. But now we remember that the system H is linear (the "L" in LTI). One of the properties of linearity is the superposition property, namely that

$$H[x_1(t) + x_2(t)] = H[x_1(t)] + H[x_2(t)]$$

Of course if the sum holds for two elements in the sum, it does for one more, and then one more, and so on, so that the property holds even for an infinite sum:

$$H\left[\sum_{k=-\infty}^{\infty} x_k(t)\right] = \sum_{k=-\infty}^{\infty} H[x_k(t)]$$

Make sure that above step makes sense to you before moving on. Now, an integral is nothing more than an infinite sum of infinitesimally skinny things, so we can use the superposition property of the system to simplify things a bit:

$$\begin{aligned} y(t) = H[x(t)] &= H \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \\ &= \int_{-\infty}^{\infty} H[x(\tau)\delta(t-\tau)]d\tau \end{aligned}$$

Remember that superposition is only part of the magic of linearity. The other linearity property is the scaling property, namely that for a linear system, $H[\alpha x(t)] = \alpha H[x(t)]$. Our system H is taking in signals that are defined in terms of time t. From the perspective of the system, $x(\tau)$ then is simply a constant, so we can continue to simplify.

$$
\begin{aligned}
y(t) = H[x(t)] &= H \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \\
&= \int_{-\infty}^{\infty} H[x(\tau)\delta(t-\tau)]d\tau \\
&= \int_{-\infty}^{\infty} x(\tau)H[\delta(t-\tau)]d\tau
\end{aligned}
$$

Hopefully the expression $H[\delta(t-\tau)]$ looks somewhat familiar...we had just been discussing the impulse response of a system, that it is the system's output when the input is a delta function: $h(t) = H[\delta(t)]$. Of course, that's not quite the same as $H[\delta(t-\tau)]$, but this is where the "*TI*" (time-invariance) of the system H being LTI comes into play. For a time-invariant system, a delay in the input produces a corresponding delay in the output, $H[x(t-t_0)] = y(t-t_0)$, or in our case, $H[\delta(t-t_0)] = h(t-t_0)$. That allows us to simplify the expression of the output even more:

$$
\begin{aligned}
y(t) = H[x(t)] &= H \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \\
&= \int_{-\infty}^{\infty} H[x(\tau)\delta(t-\tau)]d\tau \\
&= \int_{-\infty}^{\infty} x(\tau)H[\delta(t-\tau)]d\tau \\
&= \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau
\end{aligned}
$$

Any continuous-time signal can be expressed as an infinite combination of shifted delta functions. Therefore for an LTI system, the output can likewise be expressed as an infinite sum (i.e., the integral) of shifted impulse responses.

$$\delta(t) \longrightarrow \boxed{H} \longrightarrow h(t)$$

$$\delta(t-\tau) \longrightarrow \boxed{H} \longrightarrow h(t-\tau)$$

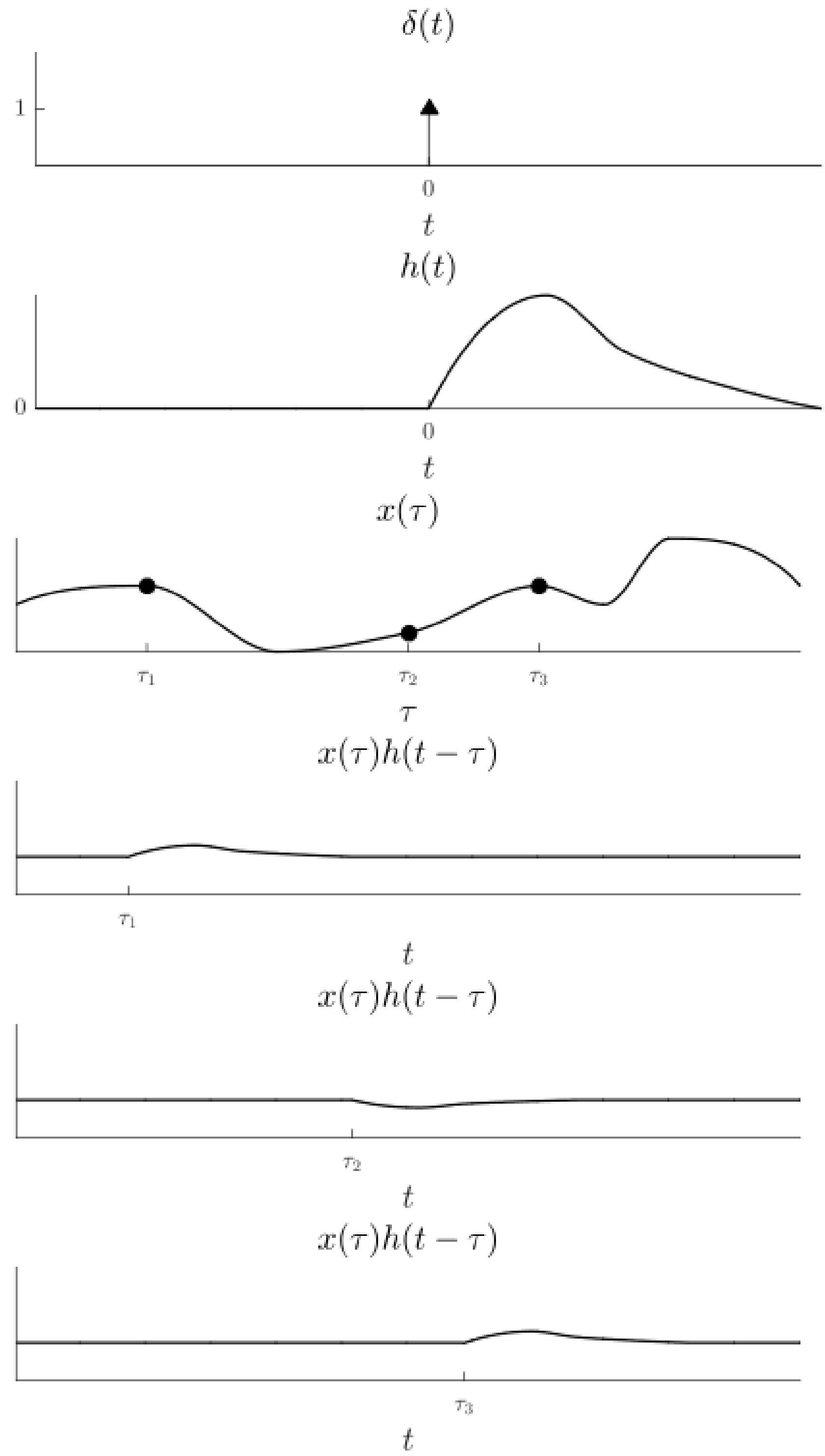
$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \longrightarrow \boxed{H} \longrightarrow \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = y(t)$$

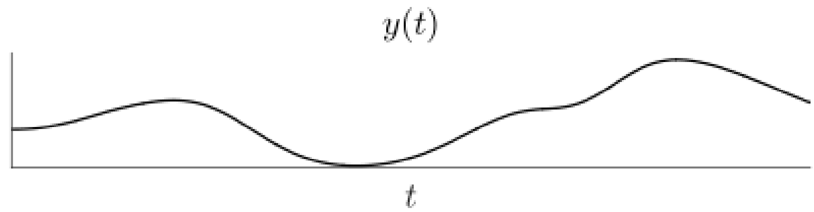
So now we see that for an LTI system, the output $y(t)$ is simply an integral involving the input $x(t)$ (although it is expressed in terms of the dummy variable τ in the integral) and the impulse response $h(t)$. At a physical level, we can think of what is happening like this: when a delta function goes through an LTI system, the output is the system's impulse response. Because we can express an input as an integral of shifted and scaled delta functions, the fact that the system is LTI means that the output is likewise an integral of shifted and scaled impulse responses. [link] shows how this looks in terms of the formulas, and [link] is an illustration of how it looks with actual signals.

So there we have it, the output of the system can be found through an integral of the input and impulse response! There will be many instances in which that will be far easier to calculate than solving a differential equation. This special integral is known as the **convolution integral** of $x(t)$ and $h(t)$, and it even has a special kind of operator, the asterisk $*$:

$$\int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = x(t) * h(t)$$

The output of an LTI system is an infinite sum (an integral) of shifted and scaled versions of the system's impulse response.





Properties of Convolution

As with other mathematical operators, convolution also has a variety of properties that will help us out as we start working with it more. Like addition and multiplication, convolution obeys the *commutative property*:

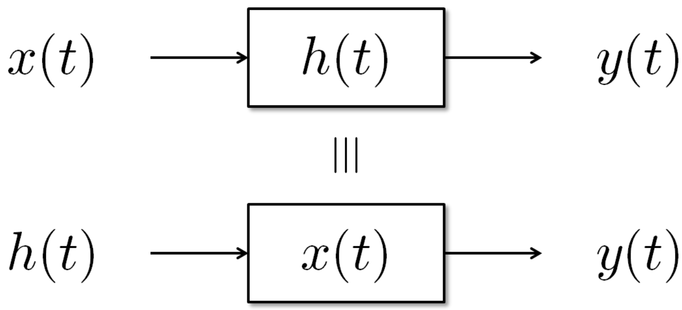
$$x_1(t) * x_2(t) = x_2(t) * x_1(t)$$

Proof of this property involves a simple change of variables, letting $u = t - \tau$ (meaning also $\tau = t - u$ and $du = -d\tau$):

$$\begin{aligned}
x_1(t) * x_2(t) &= \int_{\tau=-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau \\
&= \int_{u=t-(-\infty)}^{t-\infty} x_1(t-u)x_2(u)(-du) \\
&= -\int_{u=\infty}^{-\infty} x_1(t-u)x_2(u)du \\
&= \int_{u=-\infty}^{\infty} x_1(t-u)x_2(u)du \\
&= \int_{u=-\infty}^{\infty} x_2(u)x_1(t-u)du \\
&= x_2(t) * x_1(t)
\end{aligned}$$

When it comes to the commutative property's relevance to continuous-time LTI systems, what it means is that a signal $x(t)$ going through a system with impulse response $h(t)$ will produce the same output as a signal $h(t)$ processed by a system with an impulse response of $x(t)$ ([link]).

A graphical representation of the commutative property of convolution.

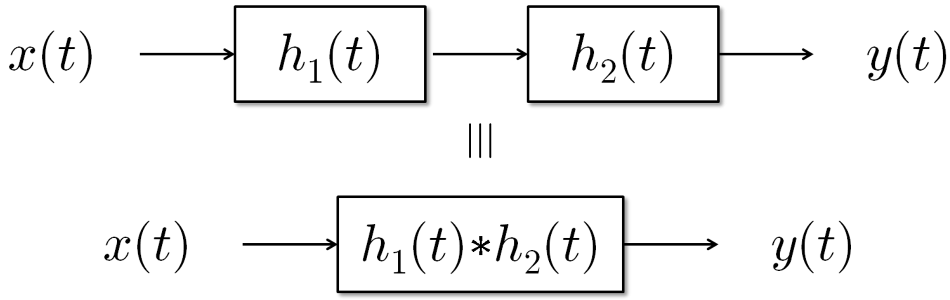


As with addition and multiplication, convolution also obeys the *associative property*, $x_1(t) * (x_2(t) * x_3(t)) = (x_1(t) * x_2(t)) * x_3(t)$. The proof of this property requires switching the order of integration and using the commutative property a couple times:

$$\begin{aligned} x_1(t) * (x_2(t) * x_3(t)) &= x_1(t) * (x_3(t) * x_2(t)) \\ &= \int_{-\infty}^{\infty} x_1(\tau_2) \int_{-\infty}^{\infty} x_3(\tau) x_2((t-\tau_2)-\tau) d\tau \, d\tau_2 \\ &= \int_{-\infty}^{\infty} x_3(\tau) \int_{-\infty}^{\infty} x_1(\tau_2) x_2((t-\tau)-\tau_2) d\tau_2 \, d\tau \\ &= x_3(t) * (x_1(t) * x_2(t)) \\ &= (x_1(t) * x_2(t)) * x_3(t) \end{aligned}$$

The consequence of this property is that two LTI systems in series can be combined to be a single system whose impulse response is the convolution of the individual systems' impulse response ([link]).

A graphical representation of the associative property of convolution.

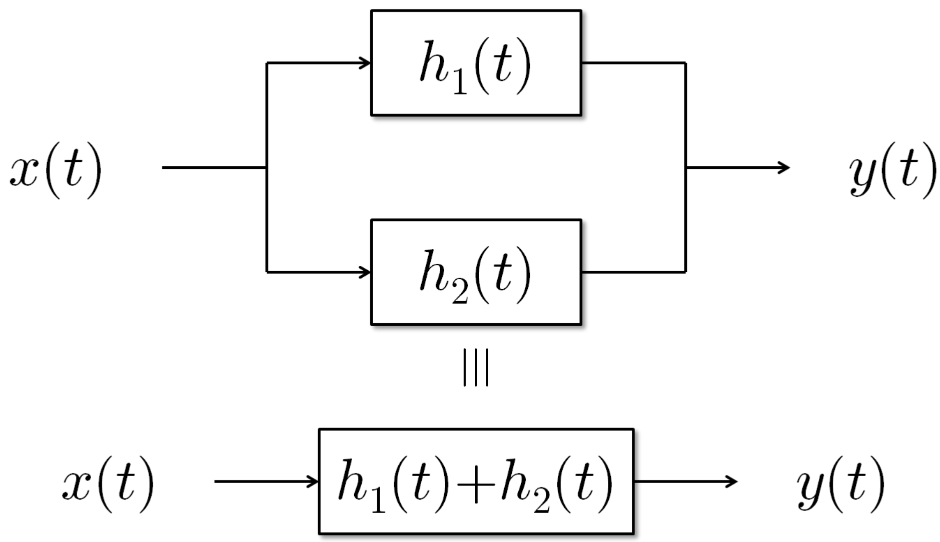


Another property, the *distributive property*, can help us to likewise simplify systems in other ways. As with multiplication and addition, the distributive property of convolution is $x_1(t) * (x_2(t) + x_3(t)) = x_1(t) * x_2(t) + x_1(t) * x_3(t)$. The proof is a matter of splitting a sum within the convolution integral into two separate integrals:

$$\begin{aligned}
x_1(t) * (x_2(t) + x_3(t)) &= \int_{-\infty}^{\infty} x_1(\tau)(x_2(t-\tau) + x_3(t-\tau))d\tau \\
&= \int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau) + x_1(\tau)x_3(t-\tau)d\tau \\
&= \int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau + \int_{-\infty}^{\infty} x_1(\tau)x_3(t-\tau))d\tau \\
&= x_1(t) * x_2(t) + x_1(t) * x_3(t)
\end{aligned}$$

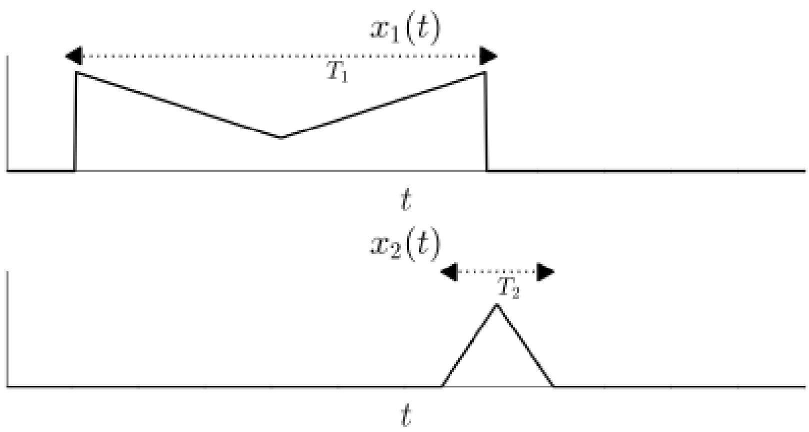
With this property, we can take two LTI systems that are in parallel and combine them into one system with an impulse response that is the sum of the individual systems' impulse responses ([link]).

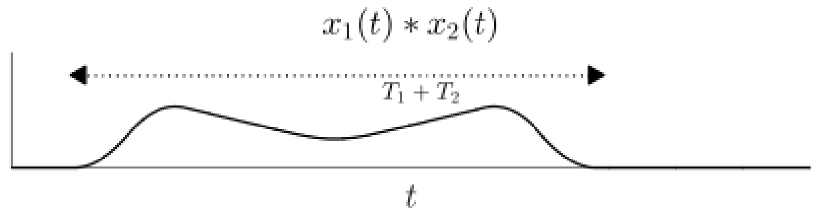
A graphical representation of the distributive property of convolution.



The final property of convolution, its *width*, has to do with the output that two convolved signals produce. Suppose a signal $x_1(t)$ is valued at 0 outside a time duration of length T_1. Suppose also a signal $x_2(t)$ is valued at 0 outside a time duration of length T_2. The convolved signal $x_1(t) * x_2(t)$ will then be valued at 0 outside a time duration of width $T_1 + T_2$ ([link]). We will see why this is the case when we begin to actually compute some convolutions.

The width of a convolved signal is the sum of the width of the two signals that were convolved to make it.




$x_1(t) * x_2(t)$
$T_1 + T_2$
t

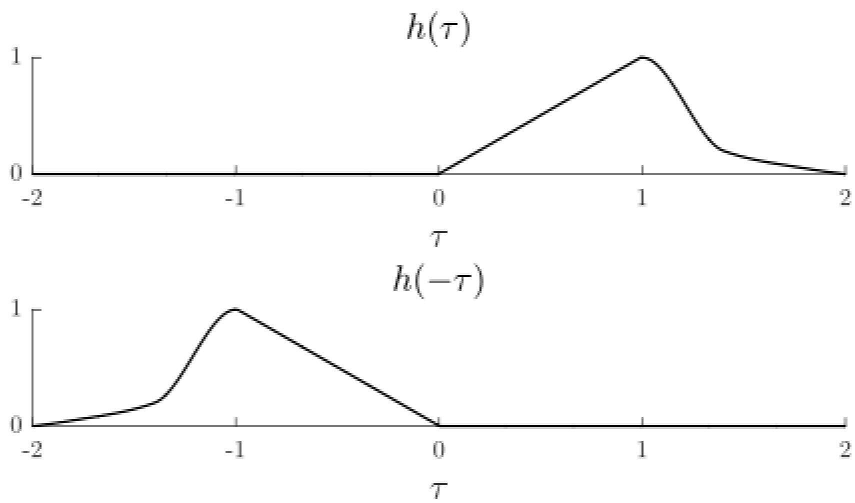
Computing Convolution

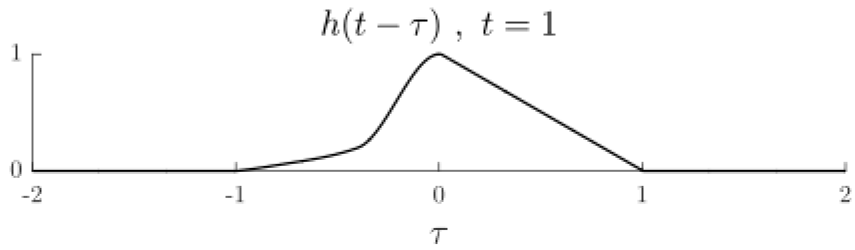
We have seen that, for LTI systems, we can find the output $y(t)$ from the input $x(t)$ and impulse response $h(t)$ through the operation of convolution $y(t) = x(t) * h(t)$, which is simply the integral:

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau.$$

In order to evaluate that integral, it can be helpful to see a graphical depiction of what it is doing. One way of looking at that integral is, as we have done previously, to see it as a combination of an infinite number of shifted and scaled impulse responses. Another way of looking at it is to consider its value for a particular time t. To do that, we'll look at the integrand. In it, one signal, $x(\tau)$ is multiplied by a "flipped and shifted" version of the other signal, $h(t-\tau)$ ([link]). Before evaluating convolution integrals, it may be helpful to first practice the "flip and shift" a bit more; see [link].

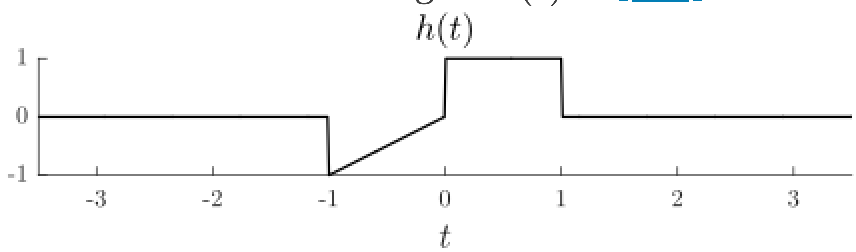
When two signals are convolved, one is "flipped and shifted" in the integrand. Suppose a signal $x(t)$ is convolved with a signal (or impulse response) $h(t)$ to produce $y(t)$. The value of $y(t)$ for each t is the integral over τ of $x(\tau)h(t-\tau)$. Above we see how $h(\tau)$ is flipped and then shifted to result in $h(t-\tau)$ (where here, $t = 1$).





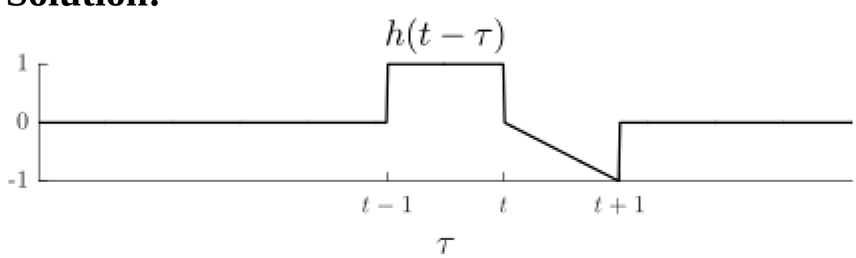
Exercise:

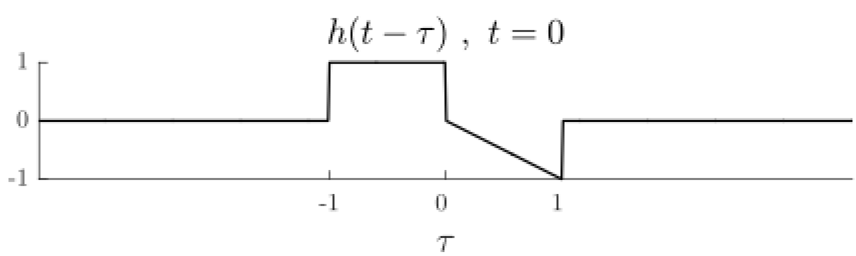
Problem: Consider the signal $h(t)$ in [link].

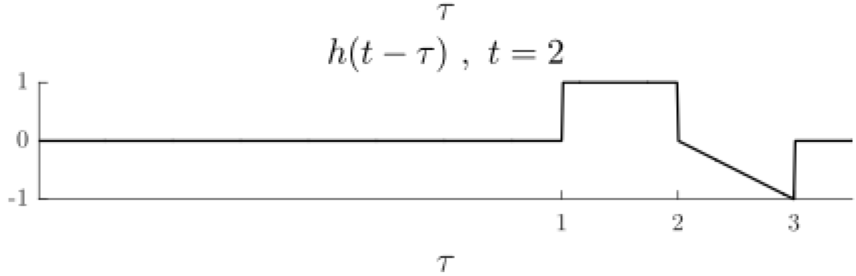


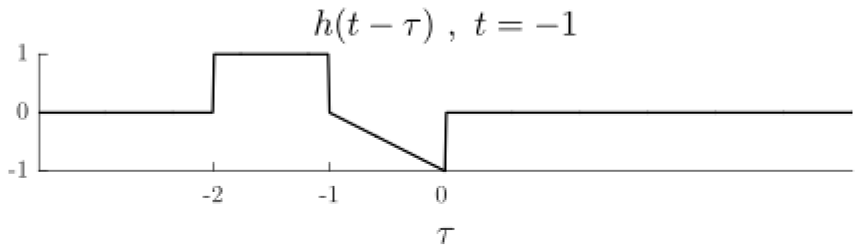
Create a labeled plot of $h(t-\tau)$ in general, and then also for $t=0$ (i.e., $h(-\tau)$), $t=2$, and $t=-1$.

Solution:



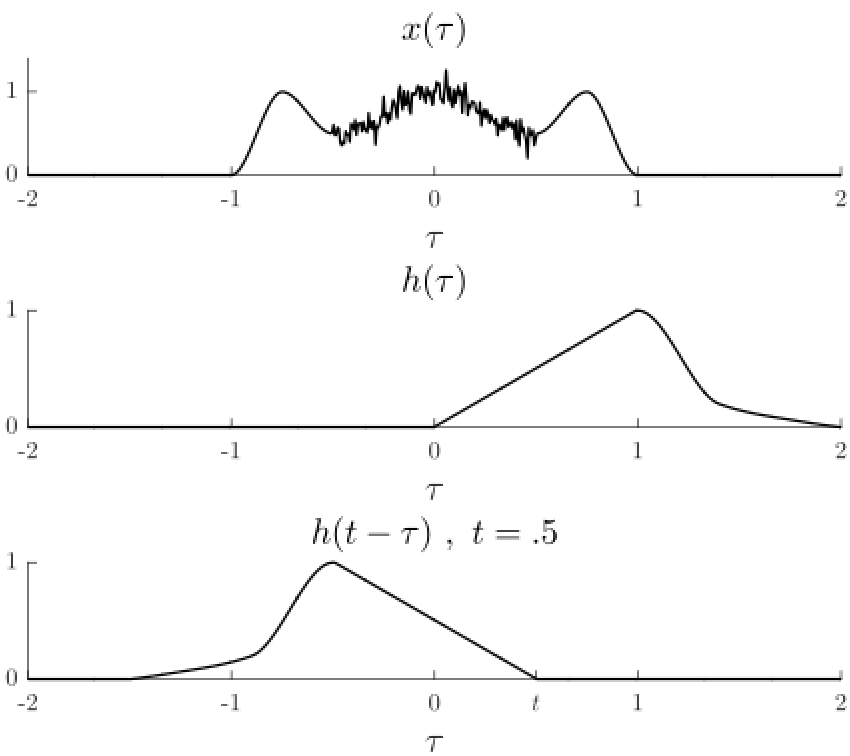


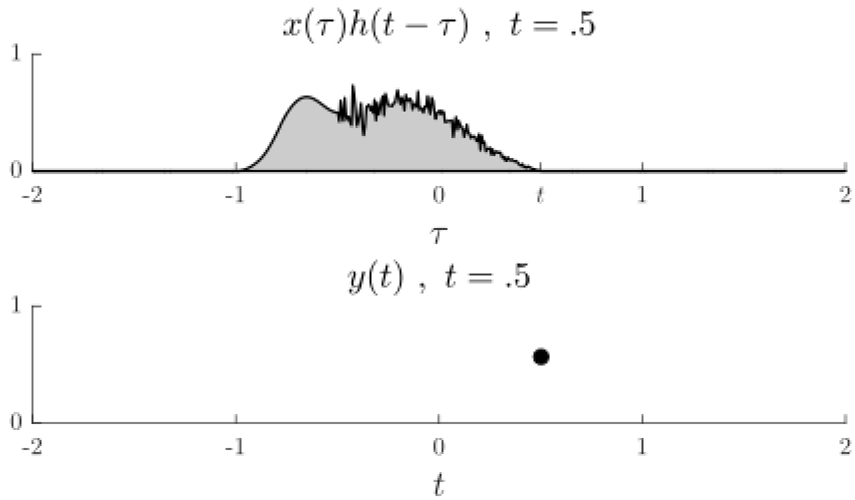




So for the convolution $y(t) = x(t) * h(t)$, one signal, $x(\tau)$ is multiplied by a "flipped and shifted" signal (or impulse response) $h(t-\tau)$, and the product is then integrated over all of τ. That produces a single numerical value for $y(t)$ at time t. The integral therefore is evaluated for *all* values of t, with $h(t-\tau)$ sliding from left to right for increasing values of t [link].

The signal $y(t)$ is the convolution of $x(t)$ and $h(t)$. $y(.5)$ is the area under the curve of $x(\tau)h(.5-\tau)$.

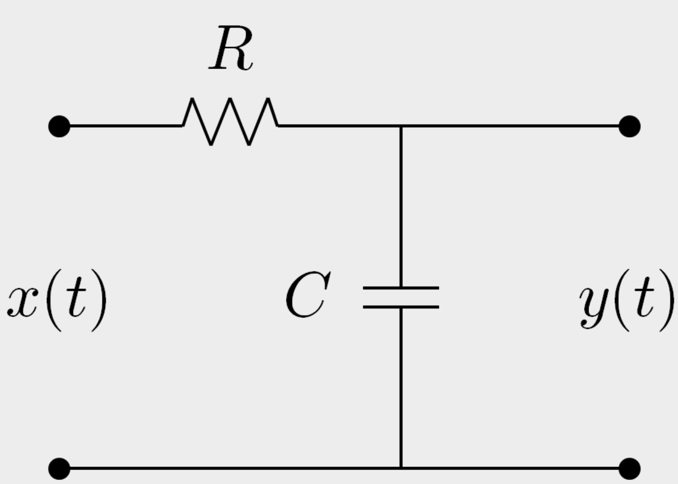




When it comes to evaluating the convolution, remember the commutative property of convolution. Since $x(t) * h(t) = h(t) * x(t)$, either $x(t)$ or $h(t)$ can be the one to flip and shift.

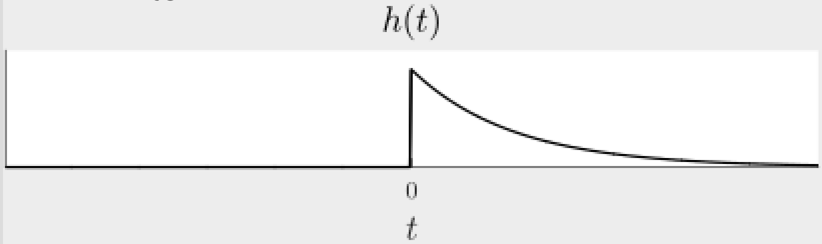
Example:
Computing Convolution

At this point we can work through calculating a convolution integral, step by step. Consider the following LTI continuous-time system, a simple RC circuit with an input of $x(t)$ and output of $y(t)$ ([link]).



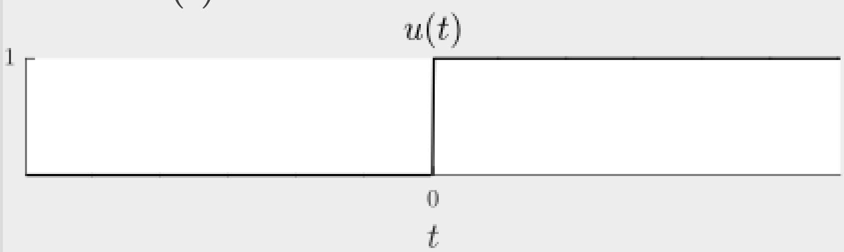
This simple RC circuit is continuous-time LTI system.

The impulse response of this system is provided for us, $h(t) = \frac{1}{RC} e^{-\frac{t}{RC}} u(t)$ ([link]).



The impulse response of an RC circuit.

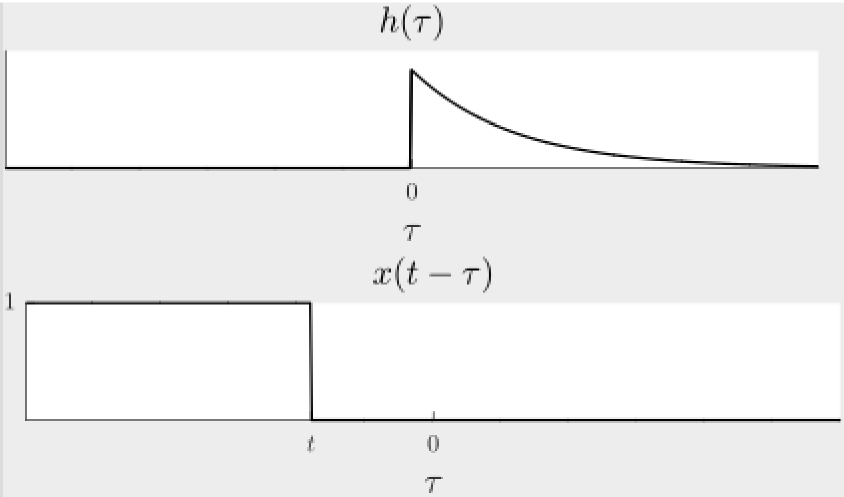
We will determine this system's output $y(t)$ when the input $x(t)$ is the step function $u(t)$.



This step function will be the input to our system.

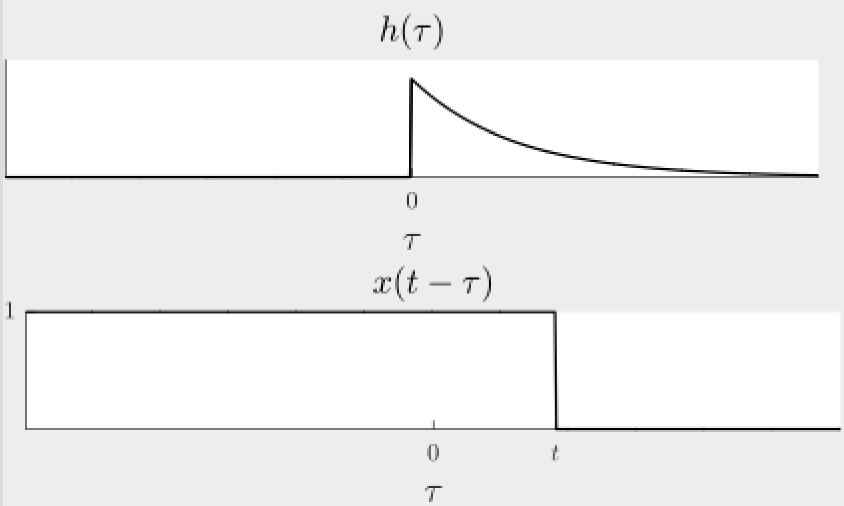
Because the system is LTI, we can find the output through convolution, $y(t) = x(t) * h(t)$:
$y(t) = \int x(\tau) h(t - \tau) d\tau$
Since convolution is commutative, we will instead choose to "flip and shift" $x(t)$, as that is a simpler signal:
$y(t) = \int h(\tau) x(t - \tau) d\tau = \int \frac{1}{RC} e^{-\frac{\tau}{RC}} x(t - \tau) d\tau$.
We will plot both $h(\tau)$ and $x(t - \tau)$ ([link]).

The product in the integrand $h(\tau) x(t - \tau)$ will be 0 for all $t < 0$

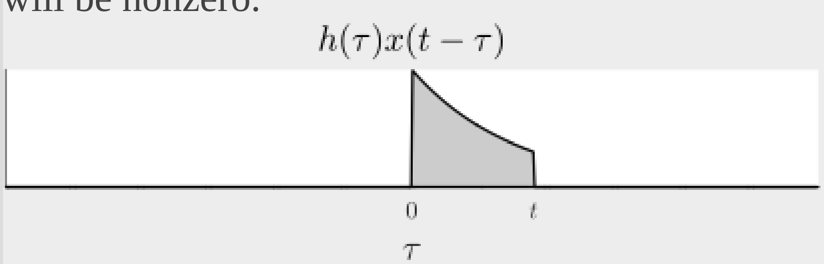


Right away, we can see that the product in the integrand $h(\tau)x(t-\tau)$ will be 0 for all $t < 0$. No matter the value of t, as long as $t < 0$, the nonzero portions of $h(\tau)$ and $x(t-\tau)$ do not overlap. Thus $y(t) = 0$ for all $t < 0$. Now we will consider the convolution for $t \geq 0$.

When $t \geq 0$, the nonzero portions of $h(\tau)$ and $x(t-\tau)$ overlap, so $y(t)$ will be nonzero.



In this case, the nonzero portions do overlap, so the convolution integral will be nonzero.



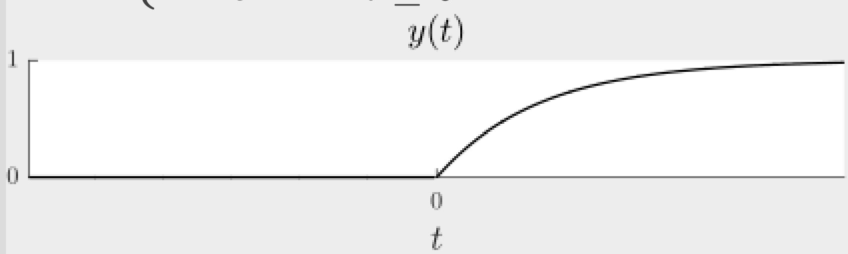
An instance of the integrand of the convolution integral for $t \geq 0$.

We can work out the value of the convolution integral for this case, $t > 0$:

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \\ &= \int_{0}^{t} \frac{1}{RC} e^{-\frac{\tau}{RC}} d\tau \\ &= \left[\frac{-RC}{RC} e^{-\frac{\tau}{RC}} \right]_0^t \\ &= 1 - e^{-\frac{t}{RC}} \end{aligned}$$

So then, we have:

$$y(t) = \begin{cases} 0 & t < 0 \\ 1 - e^{-\frac{t}{RC}} & t \geq 0 \end{cases}$$

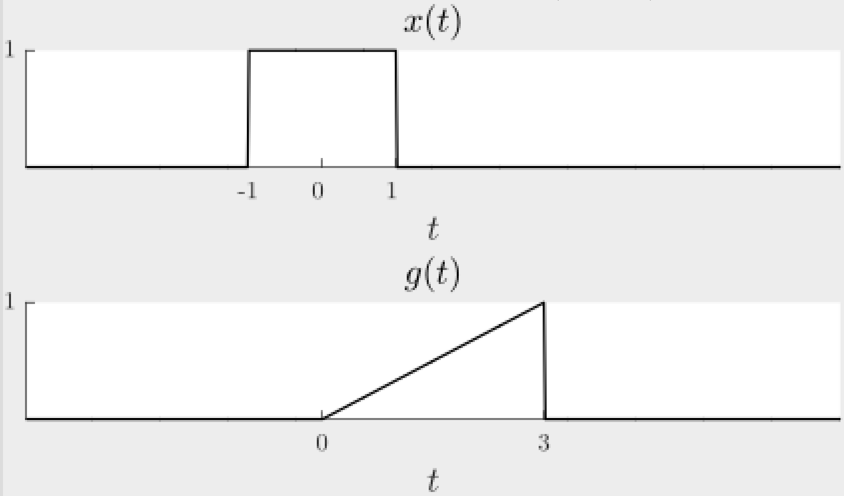


The output when a step function is input into the RC circuit.

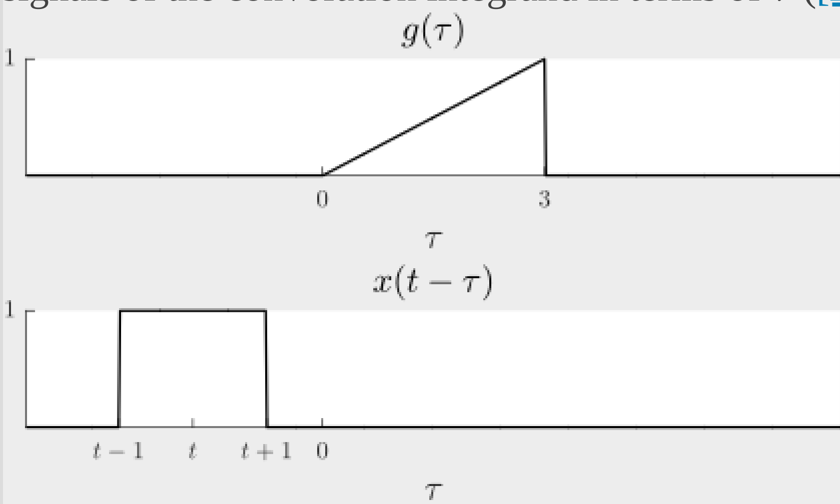
Example:
A Trickier Convolution Integral

For our first example, the convolution integral was a straightforward calculation: it was 0 for $t < 0$ and a single, simple integration for $t \geq 0$.

We'll now look at an example with a few more steps to it. Consider signals $x(t)$ and $g(t)$. $x(t)$ is 1 for $-1 \leq t \leq 1$, being 0 otherwise, and $g(t) = \frac{t}{3}$ for $0 \leq t \leq 3$, and is 0 otherwise ([link]).



We will convolve them together to produce the signal $y(t) = x(t) * g(t)$. Choosing for $x(t)$ to be the signal to flip and shift, we will next plot both signals of the convolution integrand in terms of τ ([link]).



As t (and thus, $x(t-\tau)$) moves from left to right, there will be five different cases to evaluate. The first is starting all the way at the left, when initially the signals do not overlap at all. Then the leading edge of $x(t-\tau)$ will cross over an overlap partially with $g(\tau)$. Eventually $x(t-\tau)$ will completely overlap with $g(\tau)$, before the leading edge crosses over again, until $x(t-\tau)$ is completely to the right of $g(\tau)$ and not overlapping at all. Let's now look at those in more detail.

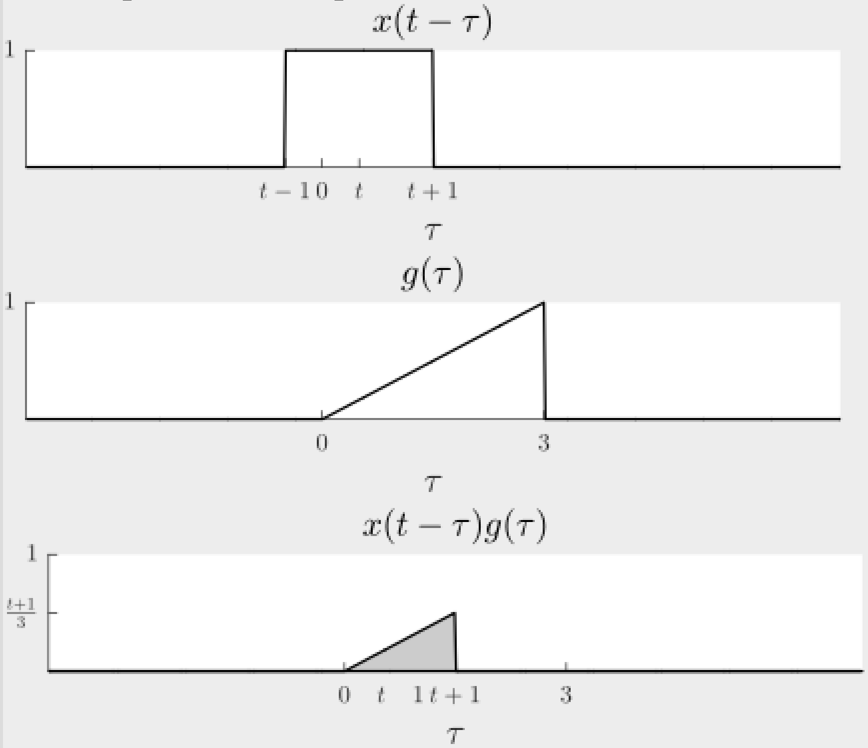
Case 1:

Notice that whenever the leading edge of $x(t-\tau)$, which is located at $\tau = t+1$, is less than $\tau = 0$, the two signals will not overlap at all, meaning the integral is zero ([link]). Simplifying $t+1<0$, we have $t<-1$. So we have

$$y(t) = \{0 \quad t<1.$$

<u>Case 2</u>:

The next case occurs when $t \geq -1$, but while the trailing edge of $x(t-\tau)$ still does not overlap. That will happen when $t-1=0$, or $t=1$ ([link]). So this partial overlap occurs for $-1 \leq t < 1$.



The convolution integral for those values of t is then:

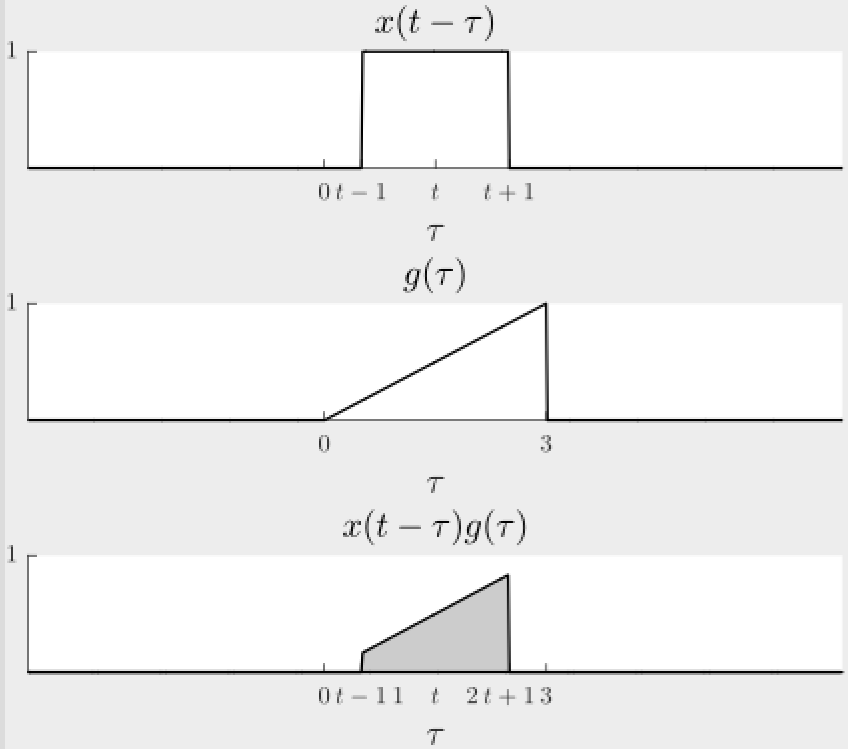
$$\begin{aligned} y(t) &= \int_0^{t+1} \frac{\tau}{3} d\tau \\ &= \left[\frac{1}{3}\frac{\tau^2}{2}\right]_0^{t+1}. \\ &= \frac{1}{6}(t^2+1) \end{aligned}$$

So far, then, we have

$$y(t) = \begin{cases} 0 & t < -1 \\ \frac{1}{6}(t^2+1) & -1 \leq t < 1 \end{cases}.$$

<u>Case 3</u>:

As t increases, the trailing edge of $x(t-\tau)$ comes within the nonzero duration of $g(\tau)$. For a time, all of $x(t-\tau)$ will overlap with $g(\tau)$. This ceases to happen when the leading edge $t+1$ is greater than 3, or when $t > 2$ ([link]). Thus the total overlap occurs for $t-1 \geq 0$ and $t < 1$, or $1 \leq t < 2$.



With this complete overlap, the integral becomes:

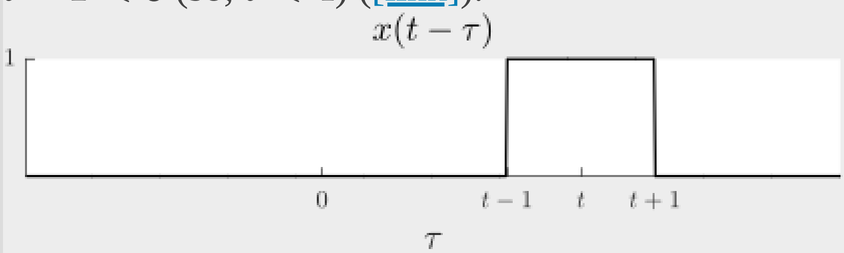
$$
\begin{aligned}
y(t) &= \int_{t-1}^{t+1} \frac{\tau}{3} d\tau \\
&= \left[\frac{1}{3}\frac{\tau^2}{2}\right]_{t-1}^{t+1} \\
&= \frac{1}{6}\left((t+1)^2 - (t-1)^2\right) \\
&= \frac{1}{6}\left(t^2 + 2t + 1 - t^2 + 2t - 1\right) \\
&= \frac{2}{3}t
\end{aligned}
$$

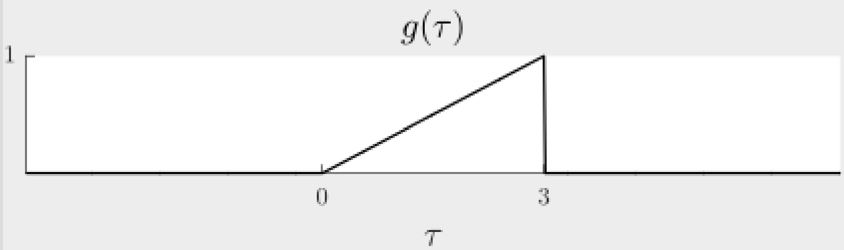
So now $y(t)$ is:

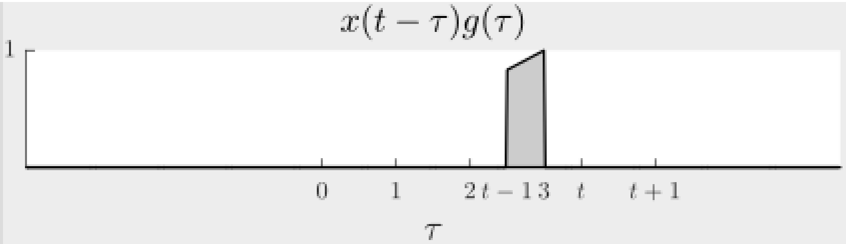
$$
y(t) = \begin{cases} 0 & t < -1 \\ \frac{1}{6}(t^2+1) & -1 \leq t < 1 \\ \frac{2}{3}t & 1 \leq t < 2 \end{cases}
$$

Case 4:

The fourth case has the leading edge of $x(t-\tau)$ moving past the edge of $g(\tau)$, but the trailing edge still overlapping, which happens for $t \geq 2$ and $t - 1 < 3$ (so, $t < 4$) ([link]).







That case gives the following convolution integral:

$$
\begin{aligned}
y(t) &= \int_{t-1}^{3} \frac{\tau}{3} d\tau \\
&= \left[\frac{1}{3}\frac{\tau^2}{2}\right]_{t-1}^{3} \\
&= \frac{1}{6}\left((3)^2 - (t-1)^2\right) \\
&= \frac{1}{6}\left(9 - t^2 + 2t - 1\right) \\
&= \frac{1}{6}(8 + 2t - t^2)
\end{aligned}
$$

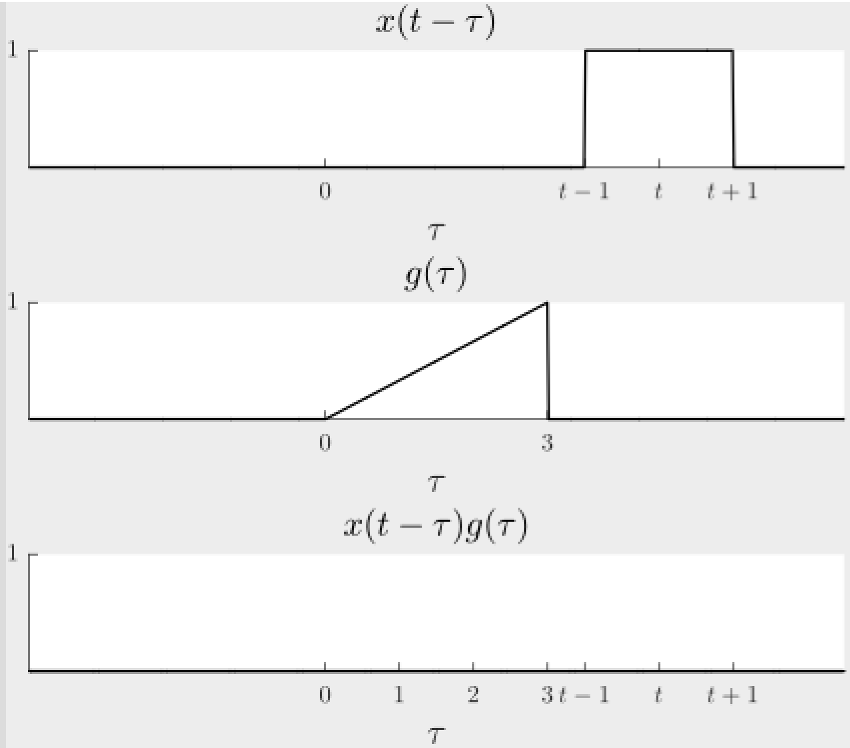
$y(t)$ now is:

$$
y(t) = \begin{cases} 0 & t < -1 \\ \frac{1}{6}(t^2+1) & -1 \leq t < 1 \\ \frac{2}{3}t & 1 \leq t < 2 \\ \frac{1}{6}(8+2t-t^2) & 2 \leq t < 4 \end{cases}
$$

Case 5:

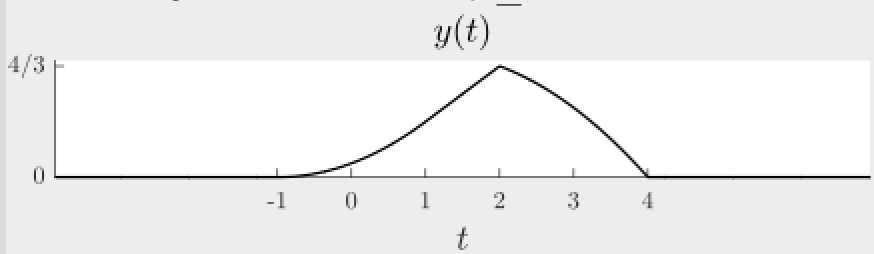
The final case is when the trailing edge $t-1$ of $x(t-\tau)$ moves past the end of $g(\tau)$, which happens for $t \geq 4$ ([link]). In this case there is no overlap, so the convolution integral is 0.

The integrand is zero for the final case in which the signals do not overlap.



That completes our expression of $y(t)$:

$$y(t) = \begin{cases} 0 & t < -1 \\ \frac{1}{6}(t^2+1) & -1 \le t < 1 \\ \frac{2}{3}t & 1 \le t < 2 \\ \frac{1}{6}(8+2t-t^2) & 2 \le t < 4 \\ 0 & t \ge 4 \end{cases}$$

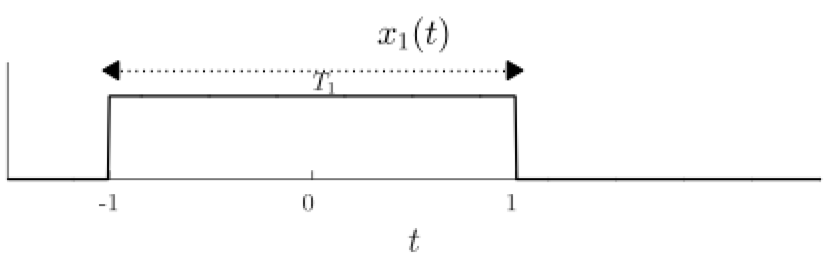


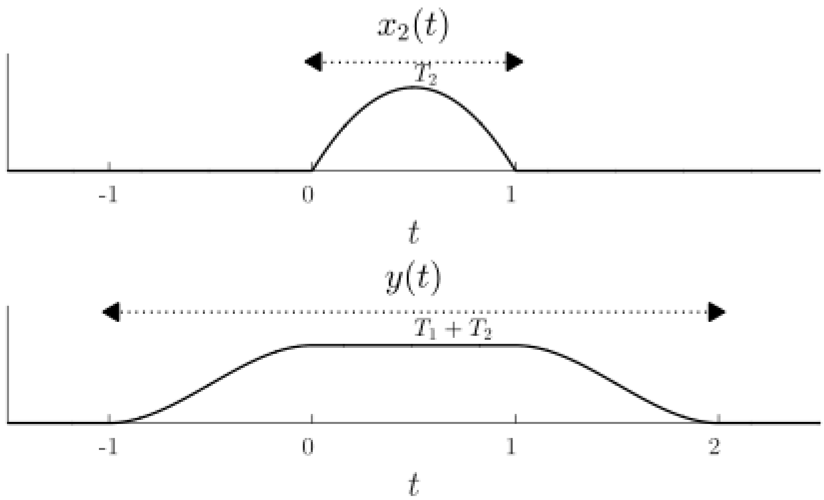
Note the non-zero duration of $y(t)$ is from $t = -1$ to $t = 4$, a duration of 5. This is, as the case for all convolution integrals, the sum of the durations of the two signals that were convolved (2 for $x(t)$ and 3 for $g(t)$).

The Width of Convolution

As we just saw in the examples above, the "width" (i.e., the continuous non-zero duration) of a convolved signal is larger than the width of each of the two signals that went into the convolution. Its width is, in fact, the sum of the two individual widths. When looking at the different cases of the convolution, we can see why this is the case. Until the leading edge of the flipped and shifted signal meets the beginning of the other signal (the transition between Case 1 and Case 2 in the example above), the convolution integral is 0. It is then (except in exceptional cases) nonzero while the flipped and shifted signal continues to travel across to the edge of the other one (the end of Case 3 above). At this point, the convolution has an output width equal to the width of the non-flipped signal. But remember there is still another non-zero case (Case 4 above), in which the rest of the flipped and shifted signal traverses past the other signal. The width of that case is equal to the width of the flipped and shifted signal. Therefore the total width is the sum of the two convolved signals' individual widths ([link]).

The non-zero duration of a convolution is equal to the sum of the durations of the two signals going into the convolution. Here $y(t) = x_1(t) * x_2(t)$, where the duration of $x_1(t)$ is and the duration of $x_2(t)$ is T_2. Suppose we choose $x_2(t)$ to flip and shift. As it crosses the $t = -1$ threshold of $x_1(t)$ the convolution begins to be non-zero, continuing across the width of $x_1(t)$, but then extending additionally for the width of $x_2(t)$ as the rest of it traverses across the $t = 1$ threshold of $x_1(t)$.





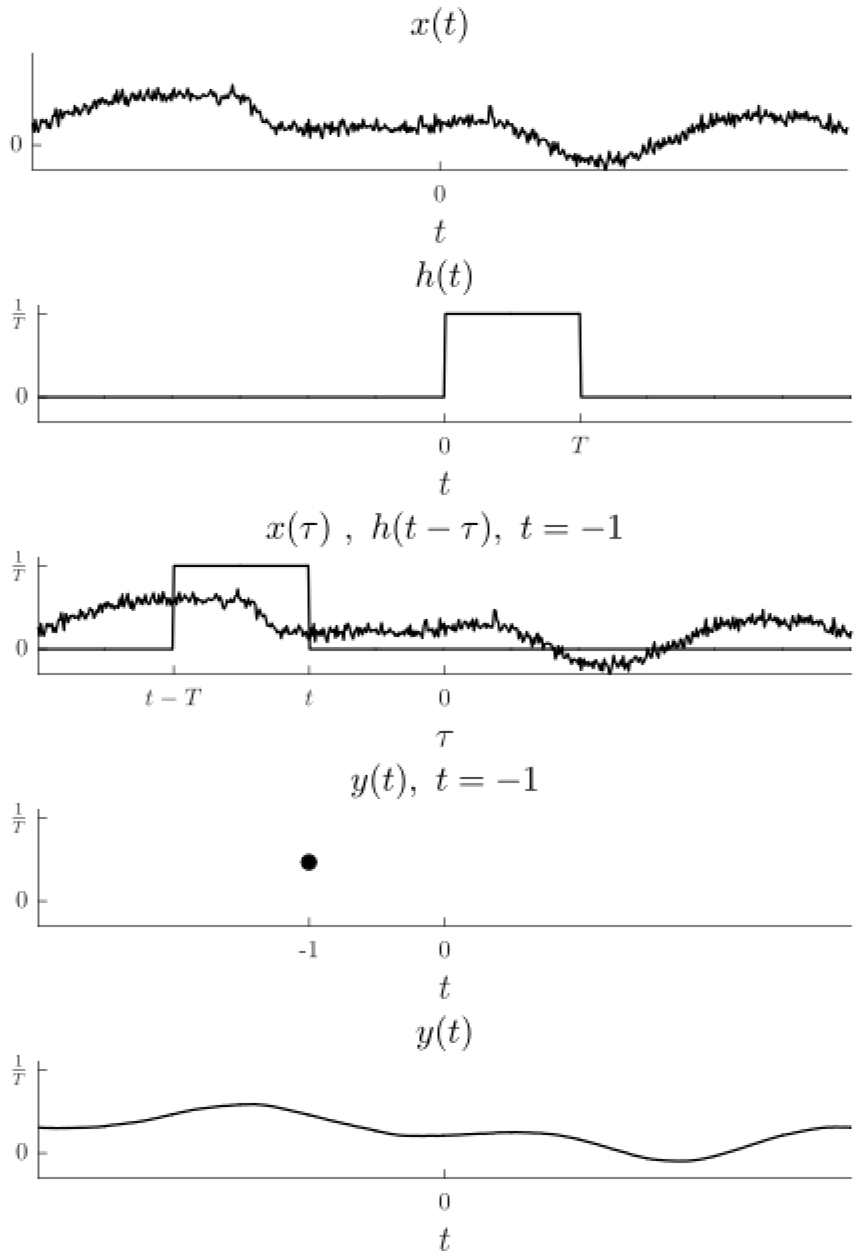
The Usefulness of Convolution

One of the reasons we study convolution is because it gives us a very straightforward way of finding the output of an LTI system (provided we know its impulse response) for a given input. But beyond that, convolution is also helpful because it provides us additional insight and intuition into what an LTI system is actually *doing*.

Recall that for an LTI system with impulse response $h(t)$, the output $y(t)$ is the convolution of the input $x(t)$ with the impulse response, $y(t) = x(t) * h(t)$. Imagine what the system is doing by flipping and shifting the impulse response. For a given time t, the system looks at a certain section of $x(t)$ (equal to the duration of the impulse response) and gives an output based upon those values being weighted by the impulse response and then integrated.

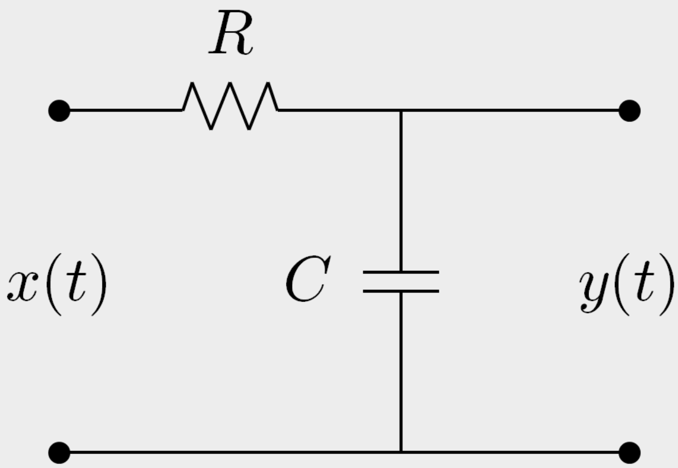
If, for example, $h(t)$ has a constant value we can see that the output at a given time is simply a scaled multiple (depending on the height and width of $h(t)$) of the mean value of $x(t)$ over the range that the flipped and shifted $h(t)$ selects ([link]). Generalizing $h(t)$, we can say that any convolution is essentially a weighted running average of the input.

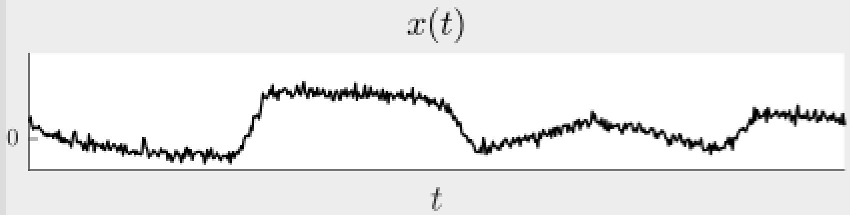
For an LTI system with the impulse response $h(t)$ shown above, we can see that for any given time t, the system produces the average value of $x(t)$ between time t and $t-T$. The output $y(t)$ is therefore a smoothed version of the input.



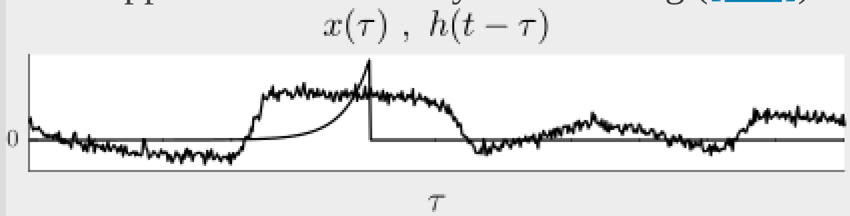
Example:

Consider an RC circuit with an input $x(t)$ as shown in [link].

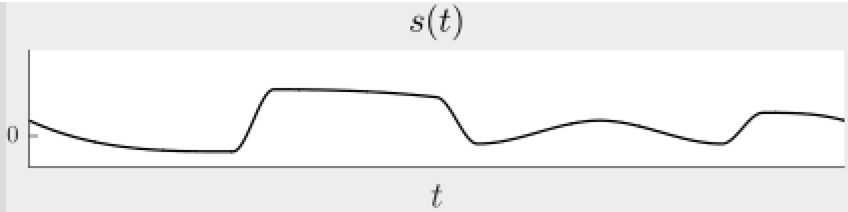


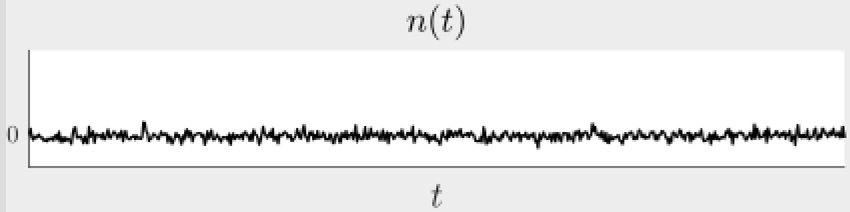


The system's circuit diagram, and perhaps even its input/output relationship--$y(t) = x(t) - RCy'(t)$--may not provide much actual intuition into how the system modifies the input signal $x(t)$. However, if we are also given the impulse response, we may get a better idea. As we have seen before, the impulse response of the RC circuit is:
$h(t) = \frac{1}{RC} e^{-\frac{t}{RC}} u(t)$.
When this impulse response is flipped and shifted across the input, we can better appreciate what this system is doing ([link]).

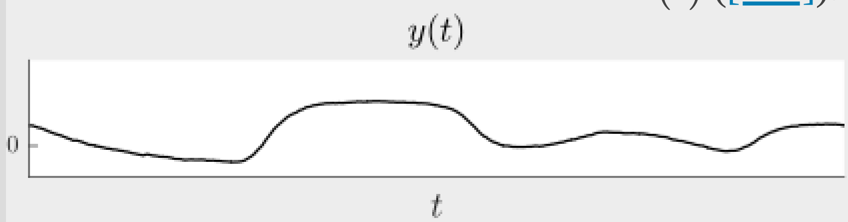


At time t, the convolution integral will weight the values of the input immediately before time t by a decaying exponential, performing a weighted average of those values (with greatest weight to the most recent one). The effect of this operation comes into even sharper relief when we recall that the convolution operation (because it represents an LTI system) is linear, and that our input appears to be the sum of a signal plus noise ([link]).





We recall that the linearity of the system means that if an input is the sum of two signals, the output will be the sum of the outputs to the individual inputs: $H[x(t)] = H[s(t) + n(t)] = H[s(t)] + H[n(t)]$. The convolution with the decaying exponential, which amounts to a running weighted average, will result in slightly smoothing out the signal $s(t)$. However, since $n(t)$ is zero-mean, the weighted averaging will actually nearly zero it out! The result then of $x(t)$ going through the system will therefore be a de-noised and smoothed out version of $s(t)$ ([link]).

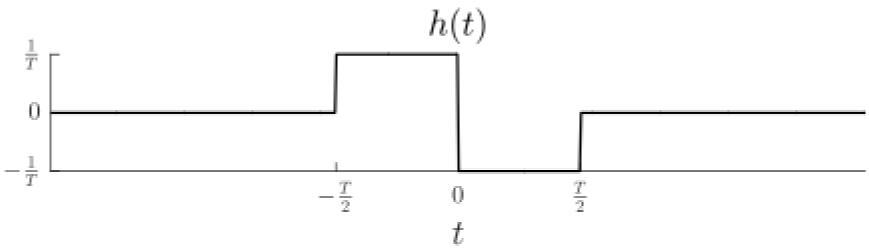


So the system acts to smooth and de-noise incoming signals. Technically it is a kind of "low pass filter," but we will learn more about that in due course.

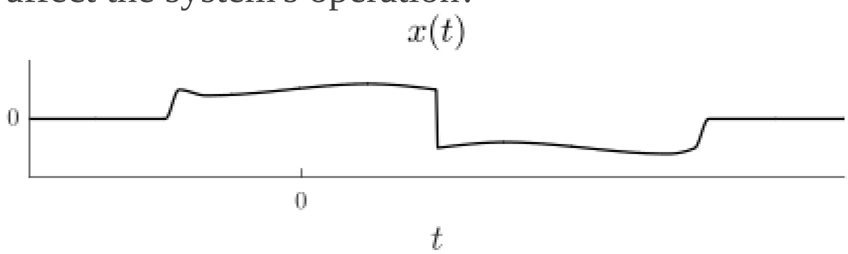
Exercise:

Problem:

Consider a continuous-time LTI system with the impulse response depicted in [link].

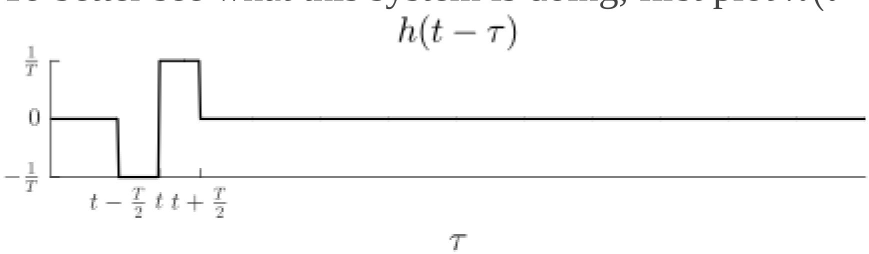


How would you describe how this system operates on input signals, e.g., the $x(t)$ in [link]? In particular, as T decreases, how does this affect the system's operation?

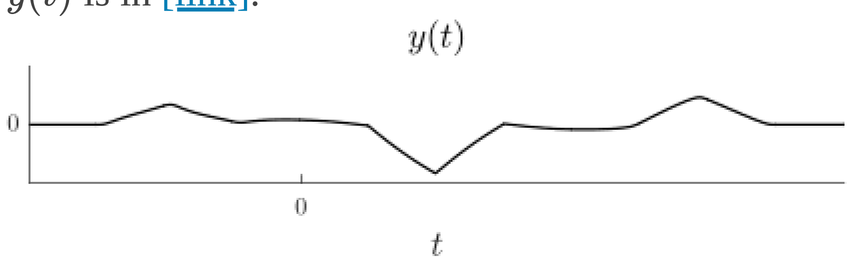


Solution:

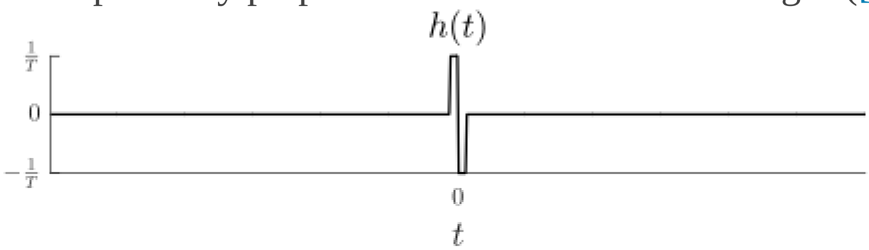
To better see what this system is doing, first plot $h(t-\tau)$ ([link]).

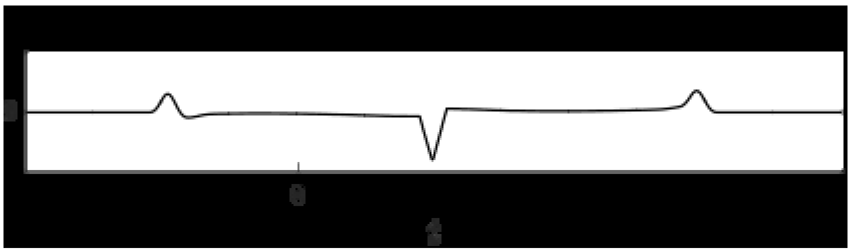


As it traverses across t , the convolution integral will essentially find the average value of $x(t)$ between t and $t+\frac{T}{2}$, and subtract from that the average value of $x(t)$ between $t-\frac{T}{2}$ and t . In so doing, it will be determining how much $x(t)$ changes right around time t . A plot of $y(t)$ is in [link].



Note how $y(t)$ is larger when $x(t)$ is increasing, and is smaller when $x(t)$ is decreasing. As T gets smaller, we would expect the system to more precisely pinpoint the locations of the changes ([link]).



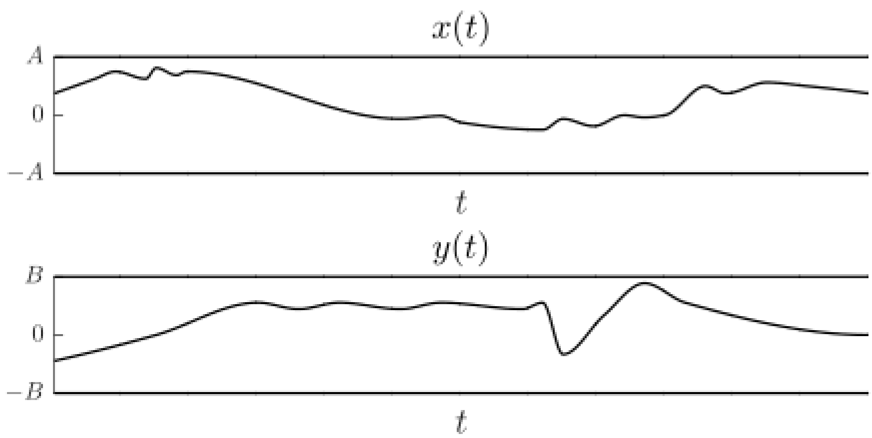


The smaller T gets, the more the system is telling us what the rate of change of $x(t)$ is immediately at time t, or in other words, the trend is to give us the derivative $x'(t)$.

Continuous-Time Impulse Response, Causality, and Stability

Recall that two properties of continuous-time systems of special interest to us are causality and system stability. A system H is causal if it only depends on present and/or past values of the input, and a system is bounded-input bounded-output (BIBO) stable if, for any bounded input $x(t)$ --meaning there exists some real number A such that $|x(t)| < A$ for all t), the output $y(t$-- will also be bounded (by some real number B ([link]).

A system is bounded-input bounded-output (BIBO) stable if, for any bounded input signal $x(t)$, the output $y(t)$ will also be bounded. Here we have an example of a bounded input, with $|x(t)| < A$, and a bounded output, $|y(t)| < B$.

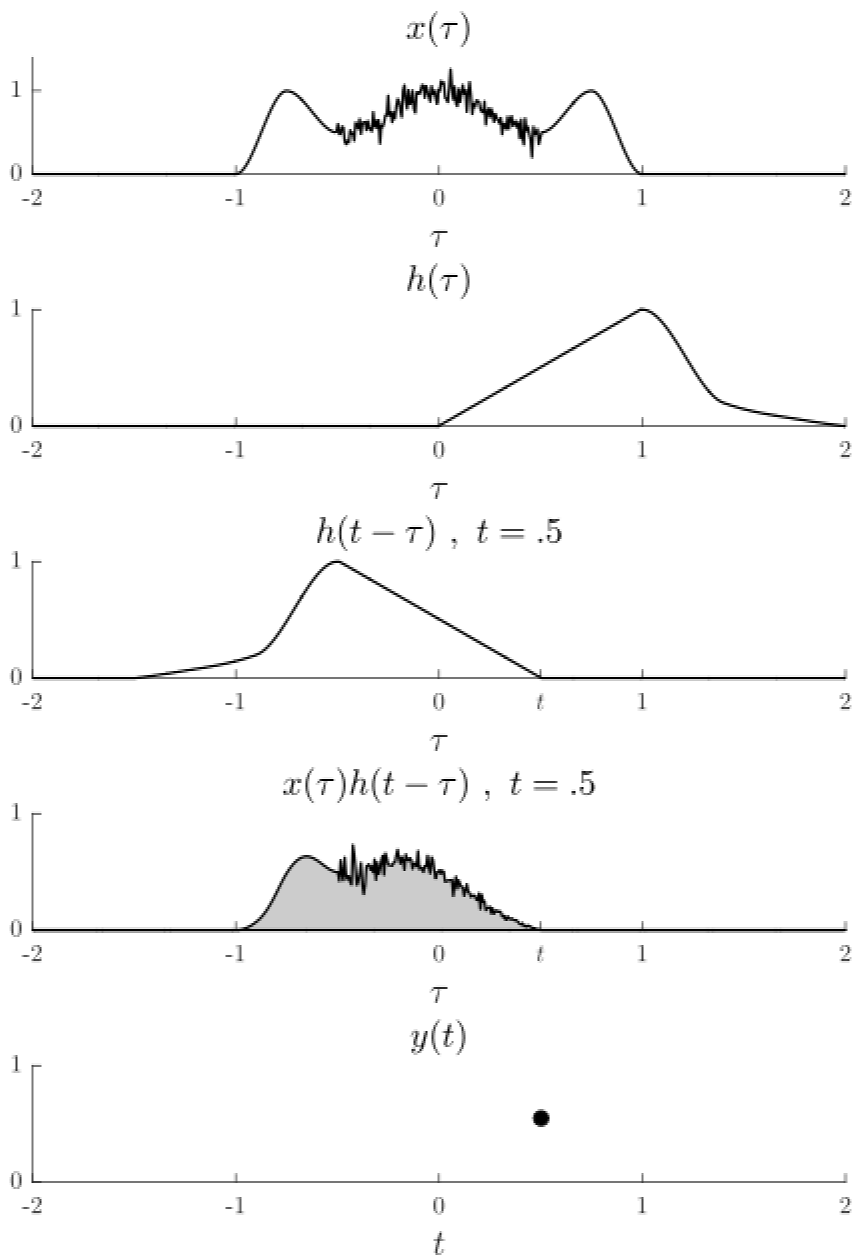


We also recall that for LTI systems, the impulse response $h(t)$ tells us everything we need to know about a system, because the output to any given input is simply the convolution of the input with $h(t)$. If indeed the impulse response completely describes an LTI system, then it follows that we ought to be able to somehow examine an LTI system's impulse response to determine whether or not the system is causal and/or BIBO stable. And it turns out, of course, that such is the case.

The Impulse Response and Causality

Let's take a look again at what convolution looks like. For an LTI system with impulse response $h(t)$, the output $y(t)$ is the convolution of the input $x(t)$ with $h(t)$. Note especially the orientation of $h(t)$ in [link].

In this convolution example, only values of $x(\tau)$ for $\tau < t$ contribute to the output at time t, because $h(\tau)$ was 0 for all $\tau < 0$.



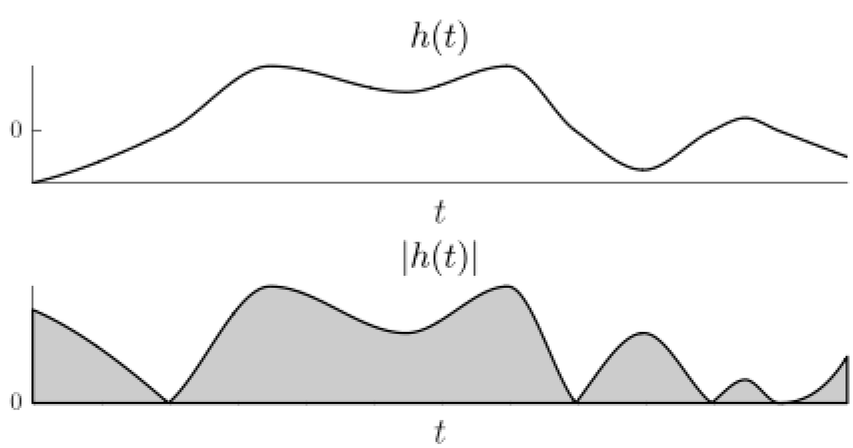
$h(\tau)$ is zero for all $t < 0$, once it is flipped, only $x(\tau)$ for $\tau \leq t$ contribute (shaded above) to the convolution integral $y(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau$. In other words, no future values of $x(t)$ go in to the computation of $y(t)$ for any given t; the system is causal!

Because of that, we can see that the impulse response does indeed tell us if a system is causal or not. Because of the way the convolution integral works, a system is causal if (and only if) its impulse response $h(t)$ is 0 for all $t < 0$.

The Impulse Response and Stability

Likewise, we can determine if an LTI system H is BIBO stable simply from considering its impulse response $h(t)$. The relationship between BIBO stability and the impulse response is as follows: An LTI system with impulse response $h(t)$ is BIBO stable if and only if $\int_{-\infty}^{\infty} |h(t)|dt < \infty$ ([link]).

For an LTI system to be BIBO stable, the area under the curve $|h(t)|$ must be finite.



Proof of this important relationship is a matter of evaluating the convolution integral (which is why we cannot stress enough that the relationship *only holds for LTI systems*, because only for those is the output the convolution of the input and the system impulse response). We will first show the "if" side of the relationship, that a system is BIBO stable if $\int_{-\infty}^{\infty} |h(t)|dt < \infty$, or in other words, that the integral condition is sufficient for BIBO stability. Suppose we have that some arbitrary input $x(t)$ is bounded, $|x(t)| < A$, and that $\int_{-\infty}^{\infty} |h(t)|dt = B < \infty$. Using the convolution integral we have:

$$\begin{aligned}
|y(t)| &= |x(t) * h(t)| \\
&= |\int_{-\infty}^{\infty} x(t-\tau)h(\tau)d\tau| \\
&\leq \int_{-\infty}^{\infty} |x(t-\tau)h(\tau)|d\tau \\
&\leq \int_{-\infty}^{\infty} |x(t-\tau)||h(\tau)|d\tau \\
&\leq \int_{-\infty}^{\infty} A|h(\tau)|d\tau \\
&\leq A\int_{-\infty}^{\infty} |h(\tau)|d\tau \\
&\leq AB
\end{aligned}$$

The other aspect of the proof, the "only if," is a little trickier. We need to show that a system is BIBO stable *only if* the impulse response satisfies the integration condition, or in other words, that the condition is necessary for BIBO stability. In order to do that, we will show that it *not* satisfying the condition implies the system is definitely not BIBO stable (if that doesn't make sense, think about it a second: if somehow the system were BIBO

stable without the integration condition, then that would mean the condition was not really necessary). To show this, we start with the assumption that $\int\limits_{-\infty}^{\infty} |h(t)|dt = \infty$ and then we input a very special input signal $x(t)$:

$$x(t) = \begin{array}{ll} -1 & h(-t) < 0 \\ 0 & h(-t) = 0. \\ 1 & h(-t) > 0 \end{array}$$

So the input is very clearly bounded (by 1), and yet the system's output to this input will be unbounded at $t = 0$:

$$\begin{aligned} y(0) &= \int_{-\infty}^{\infty} x(0-\tau)h(\tau)d\tau \\ &= \int_{-\infty}^{\infty} x(-\tau)h(\tau)d\tau \\ &= \int_{-\infty}^{\infty} |h(\tau)|d\tau \\ &= \infty \end{aligned}$$

Exercise:

Problem:

Suppose an LTI system has an impulse response $h(t) = e^{-at}u(t)$, with $a > 0$. Is this system BIBO stable?

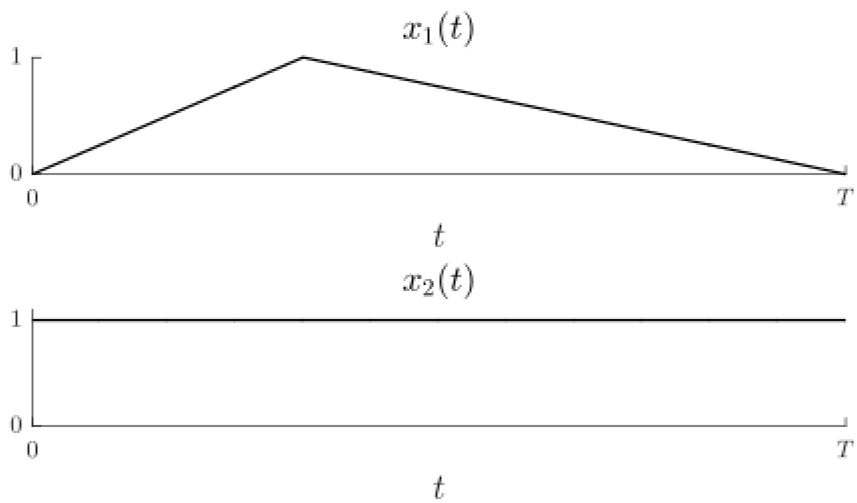
Solution:

$$
\begin{aligned}
\int_{-\infty}^{\infty} |h(t)|dt &= \int_{-\infty}^{\infty} |e^{-at}u(t)|dt \\
&= \int_{0}^{\infty} e^{-at}dt \\
&= \left. \frac{-1}{a}e^{-at} \right|_{0}^{\infty} \\
&= 0 - \frac{-1}{a} \\
&= \frac{1}{a} < \infty
\end{aligned}
$$

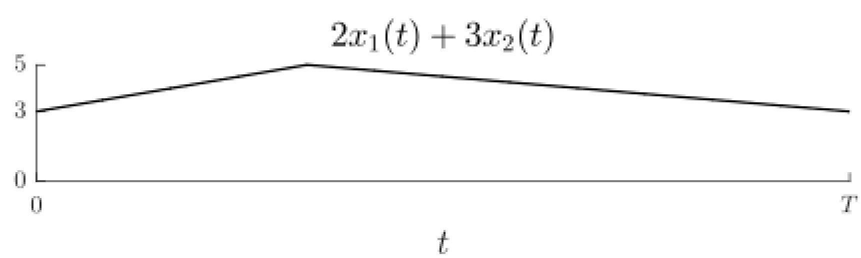
As $\int_{-\infty}^{\infty} |h(t)|dt < \infty$, the system is BIBO stable.

Linear Combinations of Signals

Consider two continuous-time finite-length (length T) signals $x_1(t)$ and $x_2(t)$ ([link]).



It is possible to combine these two signals to produce a new signal, such as $2x_1(t) + 3x_2(t)$ ([link]).

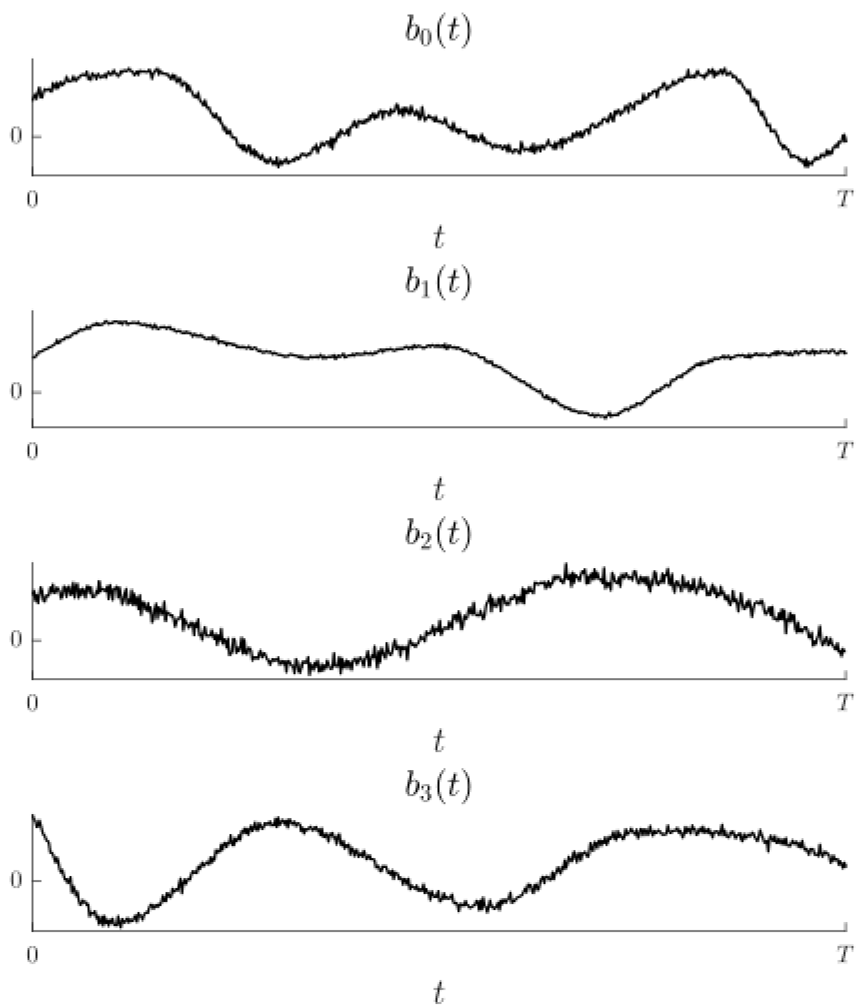


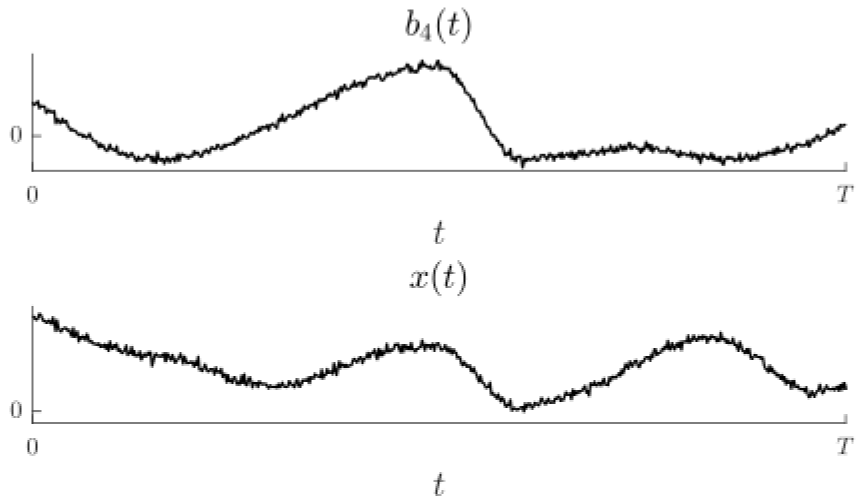
In general, if we have a collection B of N length-T signals $B = \{b_0(t), b_1(t), b_2(t), \ldots, b_{N-1}(t)\}$, we can linearly combine them in an infinite number of ways (by varying the scalar weights $c_0, c_1, c_2 \ldots c_{N-1}$) to create new signals:

$$x(t)=\sum_{n=0}^{N-1} c_n b_n(t)$$

A practical example of linearly combining signals is the mixing board one might find in a music recording studio, which takes a number of different signals (e.g., vocals, bass guitar, lead guitar, keyboard, drums, animal sounds for certain Beach Boys albums, etc.) and combines them with desired strengths to create the final track ([link]).

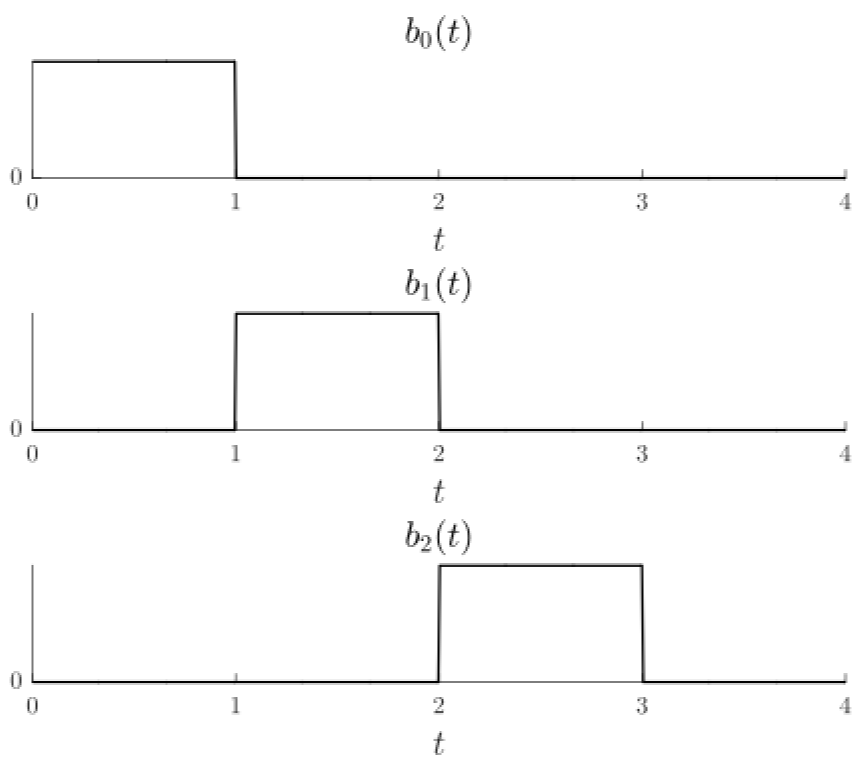
A mixing board takes signals $b_0(t) ... b_4(t)$ and combines them to produce $x(t)$.

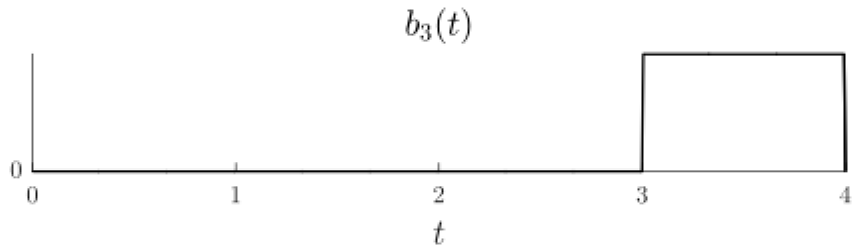




Once the set of signals B is established, all we need to know to generate different signals is the value of the weights to apply to each signal in the linear combination. Suppose that B is the set of four signals in [link].

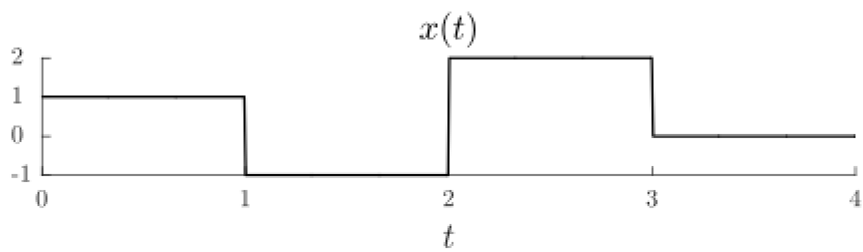
Four signals in a signal collection.





Then if we have weights of, say, $c_n = \{1, -1, 2, 0\}$, those correspond to the signal in [link].

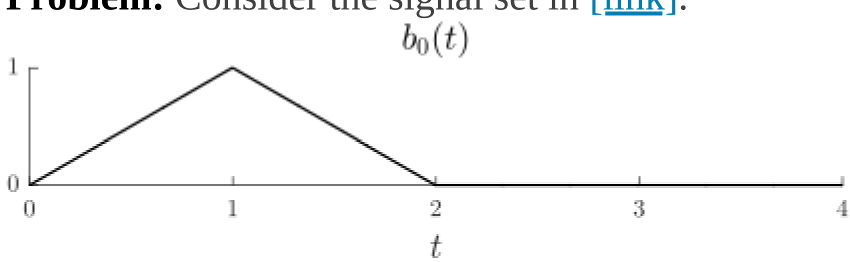
A linear combination of the signals from the signal set of [link].

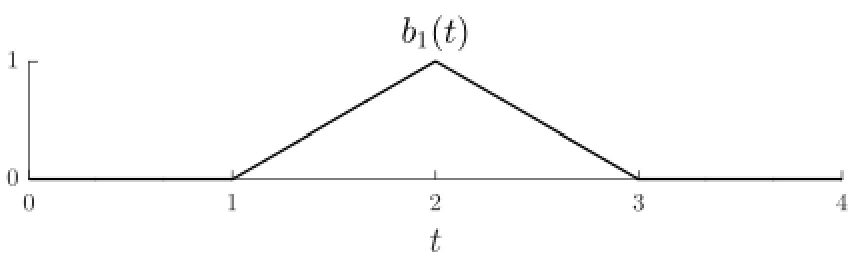


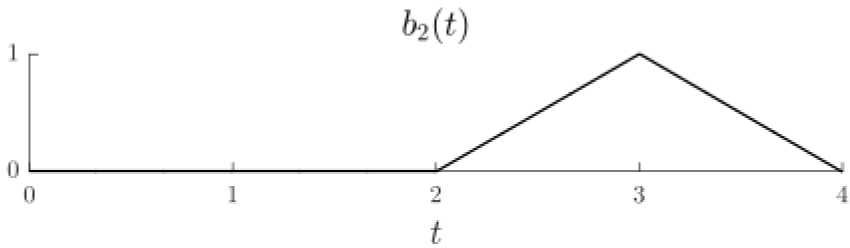
We can also work backwards. If we we know a signal $x(t)$ was made through a linear combination of signals from B, we can deduce the c_n from $x(t)$.

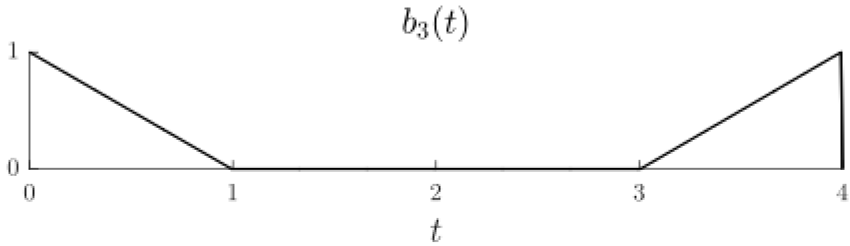
Exercise:

Problem: Consider the signal set in [link].

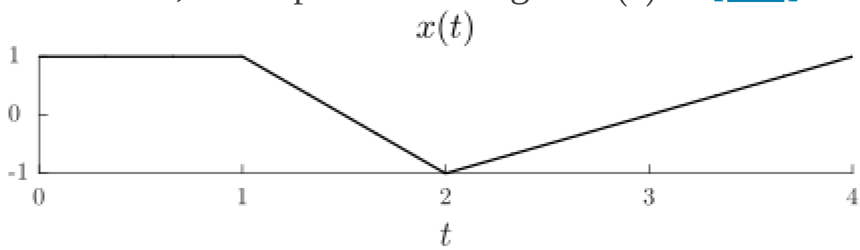








What are the weights c_n that, when used in a linear combination of the signals from the set, correspond to the signal $x(t)$ in [link]?



Solution:

$x(t)$ is valued at 1 from $t = 0$ to $t = 1$. The only way it can have that value is if $b_0(t)$ and $b_3(t)$ are summed together, so we have $c_0 = 1$ and $c_3 = 1$. Note how from $t = 3$ to $t = 4$, $x(t)$ increases from 0 to 1. As that is precisely what $b_3(t)$ is already doing there, it must be that $b_2(t)$ (the only other signal in the set nonzero from $t = 3$ to $t = 4$) does not additionally contribute, so $c_2 = 0$. Now we only need to find out how much $b_1(t)$ contributes. From $t = 2$ to $t = 3$, $x(t)$ goes from -1 to 0, while $b_1(t)$ goes from 1 to 0 (recall that $b_2(t)$ is not contributing). Thus we need $c_1 = -1$. We also verify that $c_0 b_0(t) + c_1 b_1(t)$ from $t = 1$ to $t = 2$ also matches $x(t)$, and indeed it does. So, in sum:

$c_0 = 1, c_1 = -1, c_2 = 0, c_3 = 1$

The CTFS

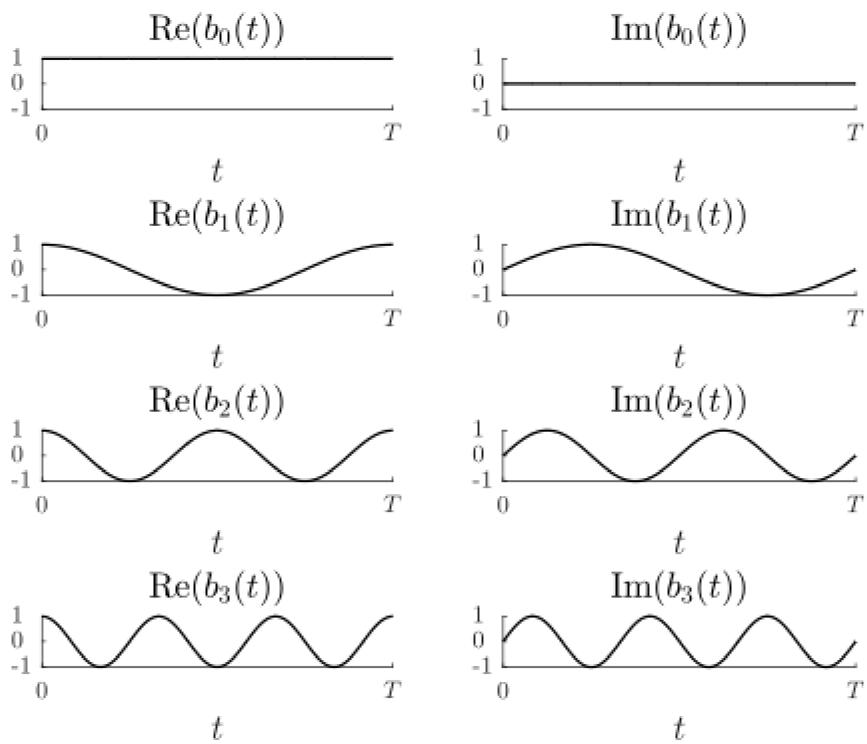
Now, even as it is clear we can vary the weights of a linear combination in an infinite number of ways--and thus create and infinite number of different signals from them--we may also ask, can we create *any* signal $x(t)$ from *any* given collection B? The answer, of course, is no. Consider again the B set from [link]. Note how each signal has a constant value between $t = 0$ and $t = 1$. Therefore it would be impossible to use them to create a new signal $x(t)$ that *did* change in value between $t = 0$ and $t = 1$.

This should not be a huge surprise--you can't make a pound cake if all you have are eggs and sugar--but it does raise the question...is there some set B that *can* create any signal $x(t)$? Or to use the culinary analogy, can we stock a pantry (signal set) so well that can be used to cook up any arbitrary dish (signal)? The surprising answer is yes, although the size of the set B will have be infinitely large.

The person credited with providing the answer to that question is Jean-Baptiste Joseph Fourier (1768-1830). He found (it took later mathematical work by others to give more rigorous proofs) that any periodic signal--or equivalently, finite-length signal--can be created with a linear combination of a (possibly infinite) number of harmonic sine waves of varying frequencies. The eventual end result is what we today call the **continuous-time Fourier series (CTFS)**. It states that any finite length (length T) signal $x(t)$--or equivalently a periodic signal with period T--can be represented with virtually perfect precision (more on that later) by a linear combination of signals from the set:

$$B = \left\{ e^{j\frac{2\pi}{T}nt} \right\}_{n=-\infty}^{\infty}$$

Plotted here are the real and imaginary parts of four of the signals from the infinitely large set that is the CTFS component signals. Note how the frequency increases as the n of $e^{j\frac{2\pi}{T}nt}$ increases.



And not only that, but there is a simple formula to determine each weight (also known as a **Fourier coefficient**) needed in the linear combination! The linear combination and that formula together are the CTFS:

- $x(t) = \sum\limits_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{t}nt}$
- $c_n = \frac{1}{T}\int\limits_{0}^{T} x(t)e^{-j\frac{2\pi}{T}nt}dt$

Take a moment to ponder the significance of the CTFS. It really is remarkable that any periodic signal can be constructed by adding up a bunch of complex sinusoids. Even a signal that looks nothing at all like a sinusoid can be expressed as an infinite sum of them. Perhaps even more impressive is the fact that--unlike as in [link] above--we don't have to guess and check what weight to assign each sinusoid; the value of the coefficient is simply the integral of the product of the signal ($x(t)$) and the complex conjugate of the sinusoid in question ($e^{-j\frac{2\pi}{T}nt}$).

A Word on CTFS Coefficients

Let's take a closer look at that operation that finds us the weight c_n corresponding to how much a particular complex sinusoid contributes to the CTFS sum. The weight c_n associated with $e^{j\frac{2\pi}{T}nt}$ is:

$$c_n = \frac{1}{T}\int_0^T x(t)e^{-j\frac{2\pi}{T}nt}dt$$

To find out how much $e^{j\frac{2\pi}{T}nt}$ contributes to the sum that forms $x(t)$, we multiply its complex conjugate by $x(t)$ and then integrate. What that operation is doing is telling us how similar $x(t)$ and $e^{j\frac{2\pi}{T}nt}$ are. If they are very similar (or precisely opposite), the value of c_n will be large, while if they are not very similar then it will be close to 0 (because, after all, if that particular $e^{j\frac{2\pi}{T}nt}$ were totally unlike $x(t)$, then it would not have anything to offer in the CTFS sum).

The technical term for that comparison integral is the signal **inner product**. You may be familiar with that term as it applies to vectors, but it is doing the same thing with length-T continuous-time signals, as well. The inner product operation on two continuous-time signals returns a (possibly complex) scalar value that indicates how alike the two signals are:

$$(f(t)|g(t)) = \int_0^T f(t)g^*(t)dt$$

If the two signals are like each other, their inner product will be large, and if they are not, it will be small. If their inner product is 0--meaning they are as unlike as can be--that special case has its own name: the two signals are **orthogonal**.

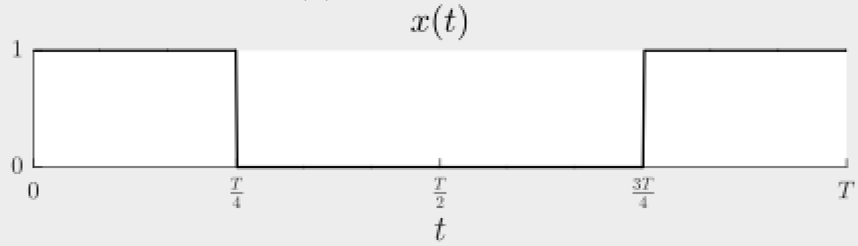
So then, the CTFS weight c_n corresponding to a particular complex sinusoid in the CTFS sum is simply the inner product of $x(t)$ with that sinusoid:

$$c_n = (x(t)|e^{j\frac{2\pi}{T}nt})$$

It's time now to get our hands dirty and actually calculate a CTFS! By that we mean the task of finding the CTFS coefficients for some signal $x(t)$.

Example:

For the square pulse signal $x(t)$ in [link], we will find the CTFS.



To find the Fourier coefficients, we'll simply follow the formula:

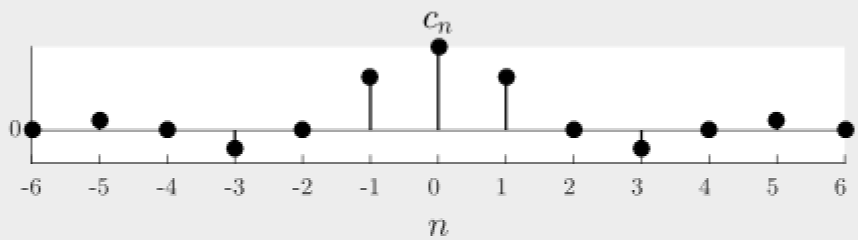
$$
\begin{aligned}
c_n &= \frac{1}{T}\int_0^T x(t)e^{-j\frac{2\pi}{T}nt}dt \\
&= \frac{1}{T}\int_0^{\frac{T}{4}} e^{-j\frac{2\pi}{T}nt}dt + \frac{1}{T}\int_{\frac{3T}{4}}^{T} e^{-j\frac{2\pi}{T}nt}dt \\
&= \frac{1}{-j\frac{2\pi}{T}Tn}\left(e^{-j\frac{2\pi}{T}\frac{T}{4}n} - 1\right) + \frac{1}{-j\frac{2\pi}{T}Tn}\left(e^{-j\frac{2\pi}{T}Tn} - e^{-j\frac{2\pi}{T}\frac{3T}{4}n}\right) \\
&= \frac{1}{j2\pi n}\left(e^{\frac{-j3\pi n}{2}} - e^{\frac{-j\pi n}{2}}\right) \\
&= \frac{1}{\pi n}\left(\frac{e^{\frac{j\pi n}{2}} - e^{\frac{-j\pi n}{2}}}{2j}\right) \\
&= \frac{1}{\pi n}\sin\left(\frac{\pi n}{2}\right) \\
&= \operatorname{sinc}\frac{\pi n}{2}
\end{aligned}
$$

As the sinc function is 0 for (nonzero) even multiples of $\frac{\pi}{2}$, we can also express the coefficients in this way:

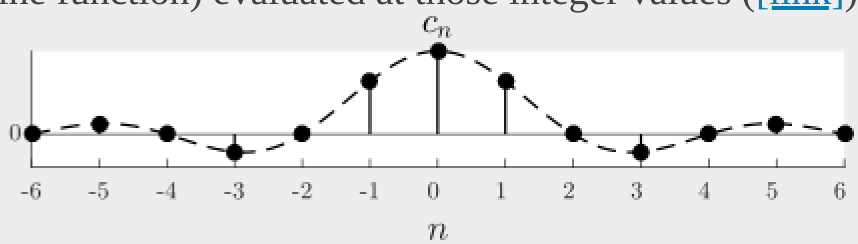
$$
c_n = \begin{cases} \frac{1}{2}, & n = 0 \\ \frac{1}{\pi n}, & n = \pm 1, \pm 5, \pm 9... \\ -\frac{1}{\pi n}, & n = \pm 3, \pm 7, \pm 11... \\ 0, & n = \pm 2, \pm 4, \pm 6... \end{cases}
$$

[link] shows a plot of the these coefficients.

The CTFS coefficients of a square pulse is a sinc function in terms of n.



Note how these coefficients are only valued for integers: $n = 0, \pm 1, = \pm 2$, etc. That said, we can still see that they are indeed a sinc function (a decaying sine function) evaluated at those integer values ([link]).



Take a second to think about how interesting a result this is. Even though $x(t)$ has sharp edges on its transition, we are able to re-create it with a sum of (smooth!) complex exponentials.

Continuous-Time Fourier Series Properties

Recall that the CTFS for a finite-length (length T) continuous-time signal $x(t)$ is defined as:

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt}$$

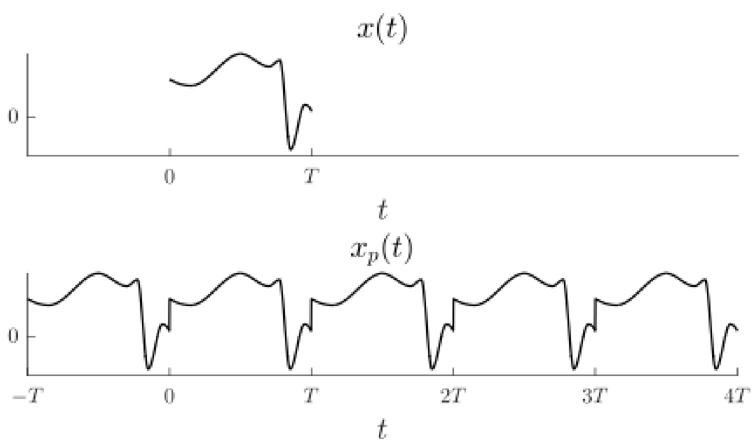
$$c_n = \frac{1}{T}\int_0^T x(t)e^{-j\frac{2\pi}{T}nt}$$

Having defined what the CTFS is, we'll now look at some if its properties.

Periodicity of the CTFS

We have said before that the CTFS applies equally to length-T finite-length signals as well as periodic signals (with a period of T), meaning that a finite-length signal $x(t)$ and the periodized version of it $x_p(t)$ would both have the exact same CTFS coefficients ([link]).

The finite-length signal $x(t)$ and its periodized version $x_p(t)$ have identical CTFS coefficients.



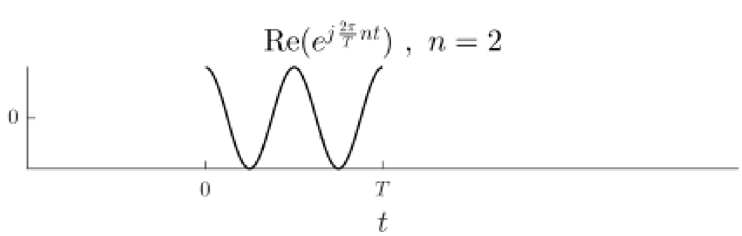
In order to see why this is so, we need only look at the complex sinusoids that are summed up in a CTFS. For length-T signals, these signals $e^{j\frac{2\pi}{T}nt}$ are only considered for the time period $t = 0$ to $t = T$. But let's see what happens if we look outside that time period, say at time $t + T$:

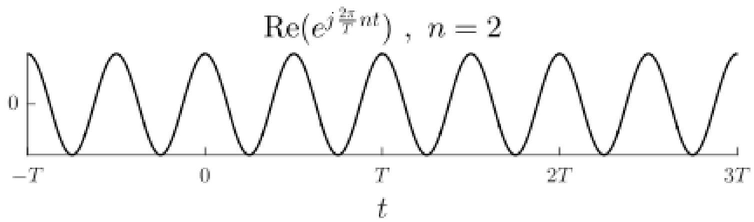
$$\begin{aligned} e^{j\frac{2\pi}{T}n(t+T)} &= e^{j\frac{2\pi}{T}nt+j\frac{2\pi}{T}T} \\ &= e^{j\frac{2\pi}{T}nt}e^{j\frac{2\pi}{T}nT} \\ &= e^{j\frac{2\pi}{T}nt}e^{j2\pi n} \\ &= e^{j\frac{2\pi}{T}nt}(1) \\ &= e^{j\frac{2\pi}{T}nt} \end{aligned}$$

All of the constituent sinusoids going into a CTFS themselves repeat every time period T ([link]). They actually repeat n times each time period T, but in any case this repetition means that the sum of all of them also repeats every time period T. As a result, even though we had originally considered the CTFS sum to represent a length-T signal $x(t)$, that sum actually gives us a T-periodic version of it if we look outside the $t = 0$ to $t = T$ time range:

$$\begin{aligned} x(t+T) &= \sum_{n=0}^{\infty} c_n e^{j\frac{2\pi}{T}n(t+T)} \\ &= \sum_{n=0}^{\infty} c_n e^{j\frac{2\pi}{T}nt+j\frac{2\pi}{T}T} \\ &= \sum_{n=0}^{\infty} c_n e^{j\frac{2\pi}{T}nt}e^{j\frac{2\pi}{T}nT} \\ &= \sum_{n=0}^{\infty} c_n e^{j\frac{2\pi}{T}nt}e^{j2\pi n} \\ &= \sum_{n=0}^{\infty} c_n e^{j\frac{2\pi}{T}nt}(1) \\ &= \sum_{n=0}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\ &= x(t) \end{aligned}$$

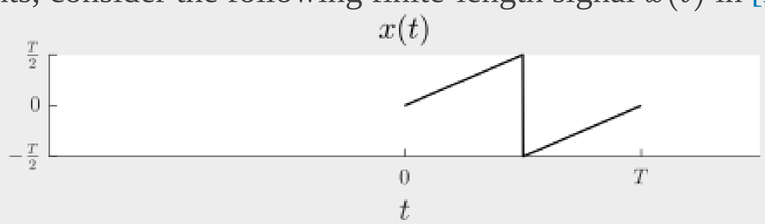
The complex sinusoids of the CTFS sum repeat every time period T.





One of helpful consequences of this finite-length/periodic relationship is that it gives us some flexibility in finding the CTFS coefficients. Recall that a finite-length signal and a periodized version of it have identical CTFS coefficients. We can therefore choose any contiguous length-T portion of the periodized signal and use the CTFS integral formula on that portion. Doing so may make the integration easier, as we will see in the following example.

Example:
To see how the finite-length/periodic relationship of the CTFS can help us in finding CTFS coefficients, consider the following finite-length signal $x(t)$ in [link].



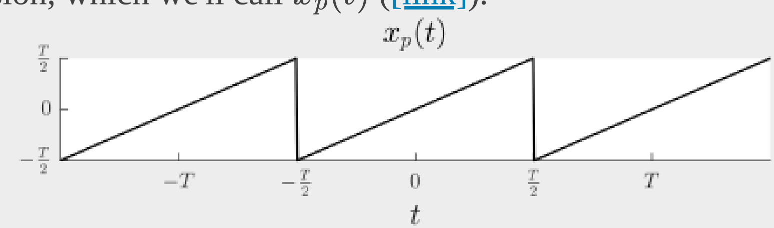
The CTFS coefficients for $x(t)$ can be found with the following formula:

$$c_n = \frac{1}{T}\int_0^T x(t) e^{-j\frac{2\pi}{T}nt} dt$$

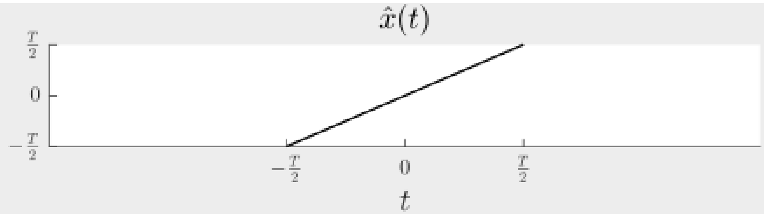
Note, however, that because $x(t) = t$ from $t = 0$ to $t = \frac{T}{2}$, and $t - T$ from $t = \frac{T}{2}$ to $t = T$, we would actually need to divide that integral into two integrals:

$$c_n = \frac{1}{T}\int_0^{\frac{T}{2}} te^{-j\frac{2\pi}{T}nt} dt + \frac{1}{T}\int_{\frac{T}{2}}^{T} (t - T) e^{-j\frac{2\pi}{T}nt} dt$$

As we have already seen, the CTFS coefficients for $x(t)$ are identical to those of its periodized version, which we'll call $x_p(t)$ ([link]).



Now consider the finite-length signal $\hat{x}(t)$ in [link].



Its periodized version is *also* $x_p(t)$, meaning that its CTFS coefficients are also identical to those of $x(t)$. The difference in this case, however, is that they can be found with just a single integral:

$$c_n = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} te^{-j\frac{2\pi}{T}nt} dt$$

We can also see from this exercise how to find the CTFS coefficients of a periodic signal of period T: we simply choose any length-T contiguous portion of it (whichever makes things easiest for us!) and calculate the CTFS integral formula over that portion.

CTFS Symmetry

Take another look at the CTFS coefficient formula for a finite-length or periodic continuous-time signal $x(t)$:

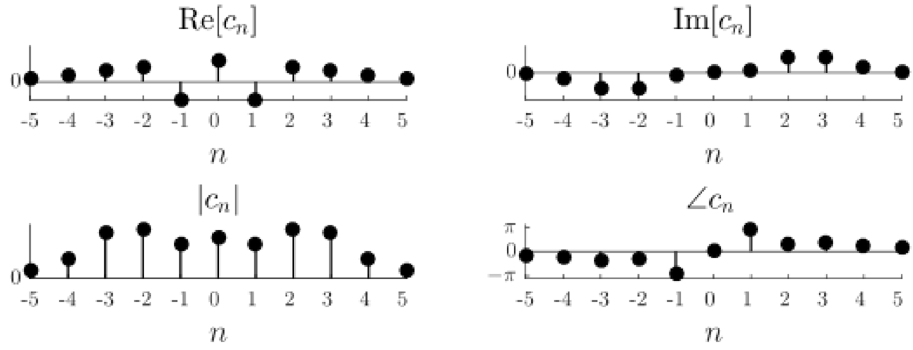
$$c_n = \frac{1}{T} \int_0^T x(t)e^{-j\frac{2\pi}{T}nt} dt$$

As the complex sinusoids $e^{-j\frac{2\pi}{T}nt}$ are conjugate symmetric (because $e^{-j\frac{2\pi}{T}nt}$ and $e^{-j\frac{2\pi}{T}n(-t)}$ are complex conjugates for all t), the CTFS coefficients will also display certain kinds of symmetry for various types of signals.

Real Signals

If a signal $x(t)$ is real, then its CTFS coefficients will be conjugate symmetric:
$c_n = (c_{-n})^*$ (also written c^*_{-n})
Conjugate symmetry implies that both the real part of the coefficients and the magnitudes of the coefficients will be even, $\mathrm{Re}[c_n] = \mathrm{Re}[c_{-n}]$, $|c_n| = |c_{-n}|$, and that both the imaginary part of the coefficients and the phases of the coefficients will be odd, $\mathrm{Im}[c_n] = -\mathrm{Im}[c_{-n}]$, $\angle c_n = \angle c_{-n}$. Thus for real signals (which are what we will be dealing with most of the time), plots of their CTFS coefficients will display this characteristic symmetry ([link]).

For real-valued signals, their CTFS coefficients will have conjugate symmetric symmetry, meaning that their real parts and magnitudes will be even, while their imaginary parts and phases will be odd.



Real and Even Signals

If a signal $x(t)$ is real and even, then its CTFS coefficients will also be real and even: $c_n = c_{-n}$, $c_n = (c_n)^*$. This makes sense because if a signal is real and even, then there cannot be any sine waves or shifted cosines or complex coefficients in the CTFS sum $x(t) = \sum c_n e^{j\frac{2\pi}{T}nt}$. With the c_n's being real and even, the CTFS sum can be simplified:

$$\begin{aligned}
x(t) &= \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\
&= c_0 + \sum_{n=-\infty}^{-1} c_n e^{j\frac{2\pi}{T}nt} + \sum_{n=1}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\
&= c_0 + \sum_{n=1}^{\infty} (c_{-n}) e^{j\frac{2\pi}{T}(-n)t} + \sum_{n=1}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\
&= c_0 + \sum_{n=1}^{\infty} c_n e^{-j\frac{2\pi}{T}nt} + \sum_{n=1}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\
&= c_0 + \sum_{n=1}^{\infty} c_n (e^{-j\frac{2\pi}{T}nt} + e^{j\frac{2\pi}{T}nt}) \\
&= c_0 + 2\sum_{n=1}^{\infty} c_n \cos(\frac{2\pi}{T}nt)
\end{aligned}$$

Since $x(t)$ is the sum of real and even signals (cosines), $x(t)$ itself is real and even.

Real and Odd Signals

For similar reasons to those for real and odd signals (the difference being that unshifted cosines are even while unshifted sines are odd), if a signal $x(t)$ is real-valued and odd, its CTFS coefficients will be imaginary and odd: $c_n = -c_{-n}$, $c_n = -(c_n)^*$.
Just as it was the case with real and even signals, the CTFS expression for real and odd signals can also be simplified, but this time in terms of sines:

$$\begin{aligned} x(t) = &\sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\ &= c_0 + \sum_{n=-\infty}^{-1} c_n e^{j\frac{2\pi}{T}nt} + \sum_{n=1}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\ &= c_0 + \sum_{n=1}^{\infty} (c_{-n}) e^{j\frac{2\pi}{T}(-n)t} + \sum_{n=1}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\ &= c_0 + \sum_{n=1}^{\infty} -c_n e^{-j\frac{2\pi}{T}nt} + \sum_{n=1}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \\ &= c_0 + \sum_{n=1}^{\infty} c_n (e^{j\frac{2\pi}{T}nt} - e^{-j\frac{2\pi}{T}nt}) \\ &= c_0 + 2j \sum_{n=1}^{\infty} c_n \sin(\frac{2\pi}{T}nt) \end{aligned}$$

CTFS and Signal Energy

To this point we have only considered the "un-normalized basis" representation of the CTFS, mostly because it is the most conventional:

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt}$$

$$c_n = \frac{1}{T} \int_0^T x(t) e^{-j\frac{2\pi}{T}nt}$$

Note the $\frac{1}{T}$ in the equation for c_n, and how it is absent in the equation for $x(t)$. It is possible to have a "normalized basis" representation of the CTFS by "spreading out" that $\frac{1}{T}$ across both equations:

$$x(t) = \frac{1}{\sqrt{T}} \sum_{n=-\infty}^{\infty} c'_n e^{j\frac{2\pi}{T}nt}$$

$$c'_n = \frac{1}{\sqrt{T}} \int_0^T x(t) e^{-j\frac{2\pi}{T}nt}$$

Expressing the CTFS in this way helps us to see of a few of its other properties more clearly.

One of the interesting properties of the CTFS (and in fact, for all of the Fourier transform relationships we will consider in the study of signals and systems) is that it preserves a signal's energy. When using the normalized CTFS, it obeys a relationship called **Plancherel's Theorem**:

$$\int_0^T |x(t)|^2 dt = \sum_{n=-\infty}^{\infty} |c_n'|^2$$

In words, what this means is that however much energy a signal had, its CTFS coefficients have the same energy! For the regular, un-normalized CTFS, the relationship simply needs to be scaled:

$$\int_0^T |x(t)|^2 dt = T \sum_{n=-\infty}^{\infty} |c_n|^2$$

It turns out this property is simply one case of a broader relationship between signals and their CTFS coefficients, a relationship expressed mathematically in **Parseval's Theorem**. Consider two signals of length T , $x(t)$ and $g(t)$. Recall their inner product, a scalar value indicating how similar (for large values) or different (for small values) they are to each other:

$$\int_0^T x(t)g^*(t)dt$$

Remarkably, Parseval's Theorem states that the CFTS preserves this relationship. The value of the signals' inner product is the same as the inner product of their CTFS coefficients!

$$\int_0^T x(t)g^*(t)dt = \sum_{n=-\infty}^{\infty} c_n' d_n'^*$$

The Plancherel theorem follows if Pareseval's theorem is applied to two identical signals.

The benefit of these two CTFS properties is that certain signal analyses--e.g., signal energy or signal inner products--can be computed either on the signals themselves, or on their CTFS coefficients. There may be cases where one calculation would be easier than the other.

CTFS Convergence

The nature of a CTFS is that we can represent some signal $x(t)$ as a (possibly) infinite sum of weighted complex exponentials. So then, suppose we have some signal $x(t)$ and find its CTFS coefficients c_n. We then create a signal $\hat{x}(t)$ using those coefficients:

$$\hat{x}(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt}$$

We would expect that $\hat{x}(t) = x(t)$; after all, it is simply the definition of the CTFS. Fourier argued that this identity always holds, i.e., that the CTFS works for all signals. But some people objected, and countered that the CTFS does not work that way for every imaginable $x(t)$. For example, the mathematician Joseph-Louis Lagrange believed the identity of the sum would only hold if $x(t)$ were a continuous function.

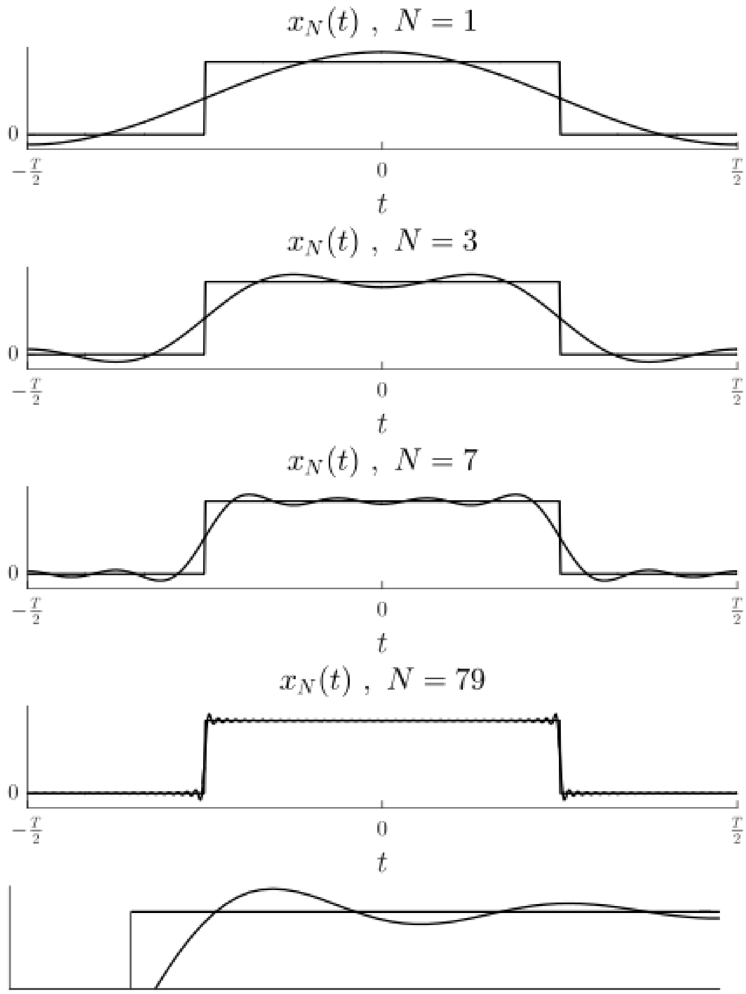
So Fourier believed the CTFS works for any signal, and Lagrange believed it would for only continuous signals. It turns out that the truth is somewhere in between, and it took the work another mathematician to prove it. Johann Peter Gustav Lejeune Dirichlet--living up to all those names--showed the identity $\hat{x}(t) = x(t)$ only holds for a certain set of conditions:

- $x(t)$ must be absolutely integral, meaning that the integral $\int_0^T |x(t)|dt$ must be finite.
- $x(t)$ must have only a finite number of local maxima and minima for any given time period.
- $x(t)$ must have only a finite number of discontinuities for any given time period.

The first condition is really the only one really keep an eye on, as the second two only apply for very strange signals (the technical term is "pathological"), such as $\sin(\frac{1}{t})$ around the time $t = 0$. But so long as those conditions are met, $\hat{x}(t) = x(t)$. The series may indeed require an infinite number of terms, but it will converge to the desired signal.

However, even if the CTFS of a signal converges, there had always been discussion of its behavior around the points of discontinuity (remember, so long as there is a finite number of these points, Dirichlet's conditions can still be satisfied). In fact, as more and more terms were added to the Fourier series of, say, a square wave, an interesting trend was noticed ([link]).

Here we have a Fourier series representation of a square wave. No matter how many terms are added to a Fourier series, there will always be ripples at the discontinuities. Here we see that even with 79 terms, when we zoom in on the discontinuity we can still see the ripple.



We see that even as the number of terms in the Fourier series increases, there will always be ripples at the discontinuities. At first it would seem that this disproves the identity $\hat{x}(t) = x(t)$, because all the Dirichlet conditions were met for the signal in the figure, and the rippled version is clearly not equal to the original. What we actually need to do is be a little more precise in defining what the equals sign means in $\hat{x}(t) = x(t)$. What it really means in the case of Fourier series is that $\| \hat{x}(t) - x(t) \|_2 = 0$. The norm of the difference in the two signals is zero. So even though there will always be a ripple, as the number of terms in the Fourier series goes to infinity, the ripple will get more and more narrow (it will always, interestingly, have the same height) and thus the area between the ripple and the original signal will tend to zero. So the ripple will never disappear, but we can make it infinitely narrow. The fact that these ripples occur at discontinuities is known as the Gibbs phenomenon.

CTFS Pairs and Linearity

There is a one-to-one relationship between a signal and its CTFS signal coefficients, meaning that a particular signal $x(t)$ has only one set of CTFS coefficients c_n , and a set of CTFS coefficients c_n corresponds to only one signal $x(t)$. As a result, we refer to a signal and its CTFS coefficients as a CTFS pair, sometimes written like this:
$x(t) \leftrightarrow c_n$

One of the helpful properties of the CTFS is that it is linear. By that we mean--just as the case with linear systems--that if we have a signal $x_1(t)$ with CTFS coefficients c_n and a signal $x_2(t)$ with coefficients d_n, with α_1 and α_2 as constants, then the following is also a CTFS pair:
$\alpha_1 x_1(t) + \alpha_2 x_2(t) \leftrightarrow \alpha_1 c_n + \alpha_2 d_n$
The proof is simply a consequence of the linearity of the integral operator in the CTFS coefficients formula:

$$\begin{aligned}\text{CTFS}\{\alpha_1 x_1(t) + \alpha_2 x_2(t)\} &= \frac{1}{T}\int_0^T (\alpha_1 x_1(t) + \alpha_2 x_2(t)) e^{-j\frac{2\pi}{T}nt} dt \\ &= \alpha_1 \frac{1}{T}\int_0^T x_1(t) e^{-j\frac{2\pi}{T}nt} dt + \alpha_2 \frac{1}{T}\int_0^T x_2(t) e^{-j\frac{2\pi}{T}nt} dt \\ &= \alpha_1 c_n + \alpha_2 d_n\end{aligned}$$

This property can sometimes make it easier to find a signal's CTFS coefficients. Suppose we know the coefficients for $x(t)$ are c_n. To find the coefficients of $2x(t)$, we do not need to recalculate the CTFS coefficient formula, they are simply $2c_n$. Or suppose we know the CTFS coefficients of two signals. The CTFS coefficients of the sum of the two signals is simply the sum of their CTFS coefficients.

CTFS Time/Frequency Relationship

Now that we have established what a CTFS pair is, we can consider some other properties. Specifically, we can look at how changes to one element of the pair affects the other. If a signal $x(t)$ is changed in some way, then how do its CTFS coefficients c_n change in response?

One way a signal $x(t)$ could be changed is by delaying it in time. Supposing that it is delayed by time t_0, such a delay will have the following effect on its CTFS coefficients:
$x(t - t_0) \leftrightarrow e^{-j\frac{2\pi}{T}t_0 n} c_n$
What happens is a phase shift in the coefficients, while the magnitude of the coefficients do not change at all. To prove this property requires a change of variables and recognition that the CTFS can be computed along any T -length section of the

signal:

$$\begin{aligned}\text{CTFS}\{x(t-t_0)\} &= \frac{1}{T}\int_{t=0}^{T} x(t-t_0)e^{-j\frac{2\pi}{T}nt}dt \\ &\text{Let } t' = t - t_0 \\ &= \frac{1}{T}\int_{t'=-t_0}^{T-t_0} x(t')e^{-j\frac{2\pi}{T}n(t'+t_0)}dt' \\ &= e^{-j\frac{2\pi}{T}t_0 n}\frac{1}{T}\int_{t'=-t_0}^{T-t_0} x(t')e^{-j\frac{2\pi}{T}nt'}dt' \\ &= e^{-j\frac{2\pi}{T}t_0 n}c_n\end{aligned}$$

Rather than apply a time shift to $x(t)$, we could instead introduce a frequency shift by multiplying it with a complex exponential $e^{j\frac{2\pi}{T}kt}$. Note the effect on the CTFS coefficients:

$$e^{j\frac{2\pi}{T}kt}x(t) \leftrightarrow c_{n-k}$$

Let's compare these two relationships:

$$x(t-t_0) \leftrightarrow e^{-j\frac{2\pi}{T}t_0 n}c_n$$

$$e^{j\frac{2\pi}{T}kt}x(t) \leftrightarrow c_{n-k}$$

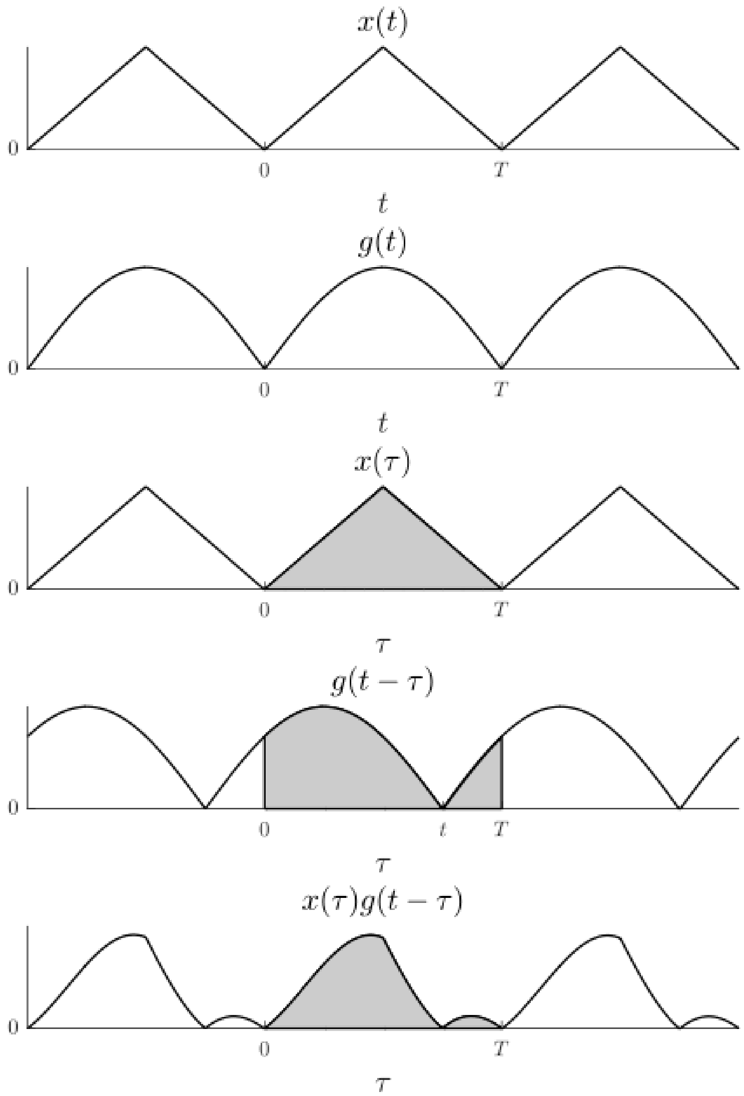
We can see here an interesting time/frequency relationship. When one side of the CTFT pair is shifted, the other is modulated. A shift in time results in modulating the CTFS by a complex exponential; a shift in frequency (i.e., in the CTFS coefficients) yields a similar modulation in time. This is only one of many corresponding relationships between time and frequency. For example, recall that a pulse signal in time has a sinc-like representation for CTFS coefficients, while a the coefficients of a sinc time signal are pulse-like.

Another time/frequency relationship involves two signals and their CTFS coefficients, and it has a special implication in the study of signals and systems. Suppose we have two periodic (or, equivalently, finite-length) signals, $x(t)$ and $g(t)$, with CTFS coefficients of c_n and d_n. The **cyclic convolution** of these signals is defined as their convolution as computed over a single period:

$$\begin{aligned}y(t) &= x(t) \circledast g(t) \\ &= \frac{1}{T}\int_0^T x(\tau)g(t-\tau)d\tau\end{aligned}$$

See [link] for an example of how it looks.

For the cyclic convolution of $x(t)$ and $g(t)$, the area under the curve $x(\tau)g(t-\tau)$ is calculated for only a single period.



What is remarkable about circular convolution is its expression with respect to CTFS pairs:

$$x(t) \leftrightarrow c_n$$

$$g(t) \leftrightarrow d_n$$

$$x(t) \circledast g(t) \leftrightarrow c_n d_n$$

The cyclical convolution of two signals in time is the same as the multiplication of their CTFS coefficients in frequency! What this means is that cyclical convolution can be computed without needing to use a convolution integral: simply multiply the CTFS coefficients of the two signals in question and then take the inverse CTFS.

Just as with time/frequency shifting, this convolution/multiplication relationship also works on the other side of the CTFT pair. Suppose rather than cyclically convolving signals $x(t)$ and $g(t)$, we instead multiply them together. The result on their CTFS coefficients will be--you guessed it--their convolution:

$x(t)g(t) \leftrightarrow c_n * d_n$

Here the convolution of the coefficients is not cyclical, and is defined as $\sum_{k=-\infty}^{\infty} c_k d_{n-k}$.

Putting this together with the cyclic convolution property, we can see that convolution in one domain (i.e., time or frequency) has the effect of multiplication in the other domain.

Summary of CTFS Time/Frequency Properties

$$x(t) \leftrightarrow c_n$$
$$g(t) \leftrightarrow d_n$$
$$x(t - t_0) \leftrightarrow e^{-j\frac{2\pi}{T}t_0 n} c_n$$
$$e^{-j\frac{2\pi}{T}kt} x(t) \leftrightarrow c_{n-k}$$
$$x(t) \circledast g(t) \leftrightarrow c_n d_n$$
$$x(t)g(t) \leftrightarrow c_n * d_n \ \left(= \sum_{k=-\infty}^{\infty} c_k d_{n-k}\right)$$

The Continuous-Time Fourier Transform

Representing Infinite-length Signals

Think about the information contained in this sentence you are reading right now. There are many ways this information could be conveyed (as the typed out text you see on your screen or book, as a handwritten note, or as a string of 1s and 0s according to the characters' ASCII encoding, or as an audio file of speech software reading the sentence...). In each case the information is the same, but the way it is conveyed can vary.

Similarly, a signal $x(t)$ can also be represented in more than one way. We can represent it in terms of its value at any given time (that's what $x(t)$ stands for, after all), but we could also represent it in other ways. To use a musical analogy, we could define a signal $x(t)$ as $x(t) = \sin(2\pi 440 t)$. That definition tells us the value of the signal at every single moment in time. But we could also define that signal in frequency: $\sin(2\pi 440 t)$ is the musical note A (in particular, the A above middle C on a piano). That is a frequency representation of $x(t)$.

Many infinite-length continuous-time signals $x(t)$ can be represented in terms of their frequency content. For finite-length signals, this representation is known as the continuous-time Fourier series (CTFS). For infinite-length signals, the representation is the **continuous-time Fourier transform** (CTFT).

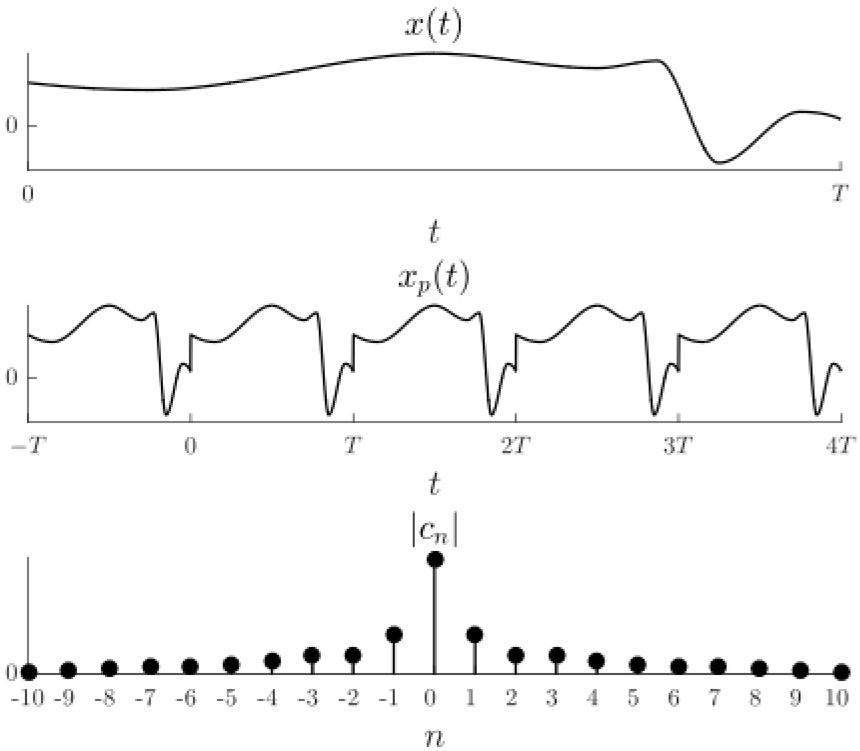
From CTFS to CTFT

Recall the definition of the CTFS. A finite-length (or periodic) continuous-time signal $x(t)$ can be represented as a (possibly infinite) weighted sum of complex sinusoids $c_n e^{j\frac{2\pi}{T}nt}$:

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt}$$

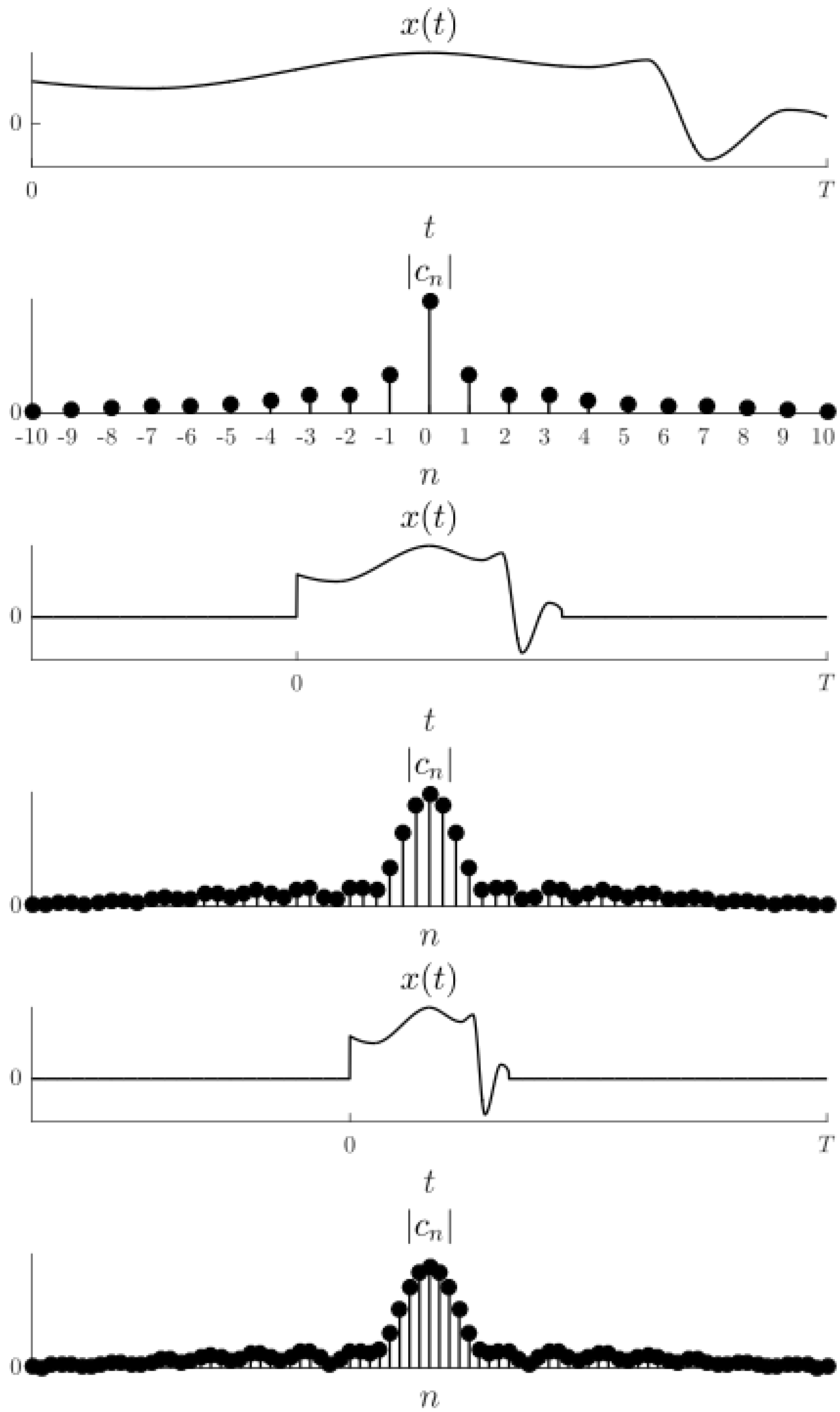
$$c_n = \frac{1}{T}\int_0^T x(t) e^{-j\frac{2\pi}{T}nt} dt$$

Finite-length signals ($x(t)$ above) or periodic signals ($x_p(t)$) can be represented as a weighted sum of complex sinusoids $c_n e^{j\frac{2\pi}{T}nt}$.



So the CTFS is a way to represent finite-length or periodic signals in terms of other signals (complex sinusoids), but what we would like to have is a similar representation for infinite-length signals. To see how that representation might look, let's see what happens to the CTFS coefficients for a finite-length signal when we make it longer and longer by adding $x(t) = 0$ space each side of it ([link]).

As our finite-length signal $x(t)$ gets larger and larger (by adding more zero space), we can see that the CTFS resolution gets finer.



What is happening as we make the signal longer and longer is that we are merely increasing the size of T in the CTFS formulas, while keeping everything else in them the same. Let's look at the formula for the CTFS coefficients:

$c_n = \frac{1}{T}\int_0^T x(t)e^{-j\frac{2\pi}{T}nt}dt$

Pay close attention to the $\frac{2\pi}{T}n$ in this formula. Note that over the course of the CTFS coefficients, it ranges from $-\infty$ to ∞ at $\frac{2\pi}{T}$ steps. As T gets larger and larger, it will still range from $-\infty$ to ∞ , but the step size gets smaller and smaller. As T goes to infinity, $\frac{2\pi}{T}n$ becomes a continuous variable going from $-\infty$ to ∞ . We'll call this variable ω, and the "coefficients" term c_n will change to $X(j\omega)$:

$X(j\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$

There are couple of things to note here. First, we've taken away the $\frac{1}{T}$ term for the time being (it will show up in the second formula), and second, we have put a j in the notation $X(j\omega)$, so as to differentiate it from a discrete-time version of the transform.

Now let's look at the second formula, the one that reconstituted the signal with the CTFS:

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt}$$

We recall that in the formula for c_n we left out the $\frac{1}{T}$, so we'll add that back in:

$$x(t) = \sum_{n=-\infty}^{\infty} \frac{1}{T} c_n e^{j\frac{2\pi}{T}nt}$$

And just to make it consistent with the rest of the formula, we'll make it $\frac{2\pi}{T}$ by balancing it with a $\frac{1}{2\pi}$ to the front of the formula, and put it after the complex sinusoid:

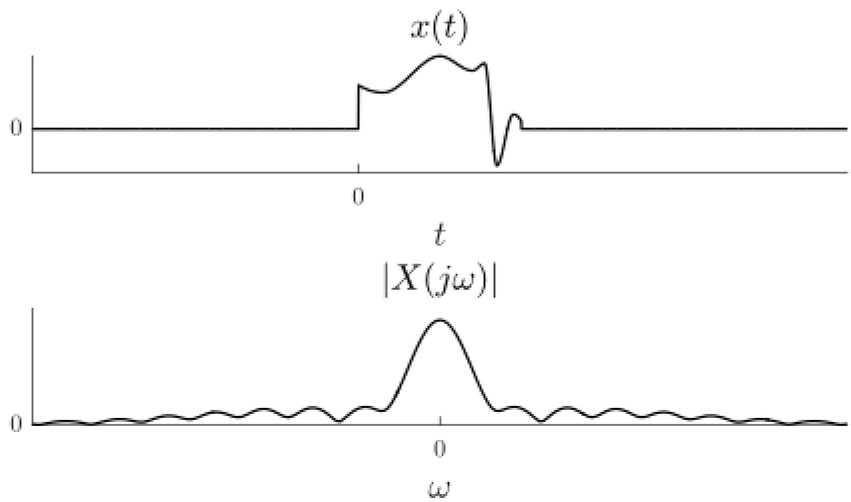
$$x(t) = \frac{1}{2\pi}\sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt}\frac{2\pi}{T}$$

Now as T goes towards infinity, recall that the $\frac{2\pi}{T}n$ becomes a continuous variable ω that goes from $-\infty$ to ∞ . The $\frac{2\pi}{T}$ term will become infinitesimally small--we'll call it $d\omega$ --and we'll no longer have a sum going in discrete units of n from $-\infty$ to ∞ , but rather an integral with ω moving continuously from $-\infty$ to ∞:

$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\omega)e^{j\omega t}d\omega$

So the main thing happening in our change from CTFS to CTFT is that the $\frac{2\pi}{T}n$ term, which traverses from $-\infty$ to ∞ at $\frac{2\pi}{T}$ increments, becomes a continuously varying term ω . This has the additional effect that the synthesis term building up a signal $x(t)$ from weighted complex sinusoids goes from being a sum to an integral.

As a signal $x(t)$ becomes infinite in length (only a portion of it is shown), it produces a continuous-valued frequency representation $X(j\omega)$.



The CTFT

So, based upon our extension of the CTFS to apply it to infinite-length signals, we have our definition of the CTFT for a signal $x(t)$:

$$X(j\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$$

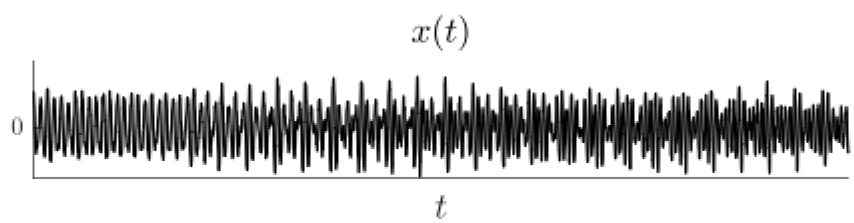
$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\omega)e^{j\omega t}d\omega$$

We can refer to the $X(j\omega) = \cdots$ formula as CTFT **analysis**, because it gives us a frequency analysis of $x(t)$, letting us know how much of each

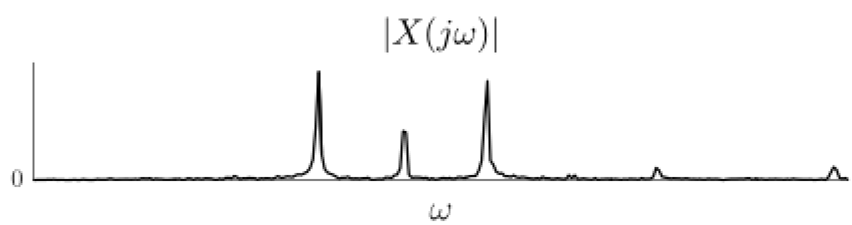
frequency ω goes into the signal $x(t)$. The $x(t) = \cdots$ formula is CTFT **synthesis**, because with it we can put $x(t)$ back together from $X(j\omega)$.

To see how the CTFT gives us an analysis of a signal, we'll pick up our musical example again. Suppose a major chord is played on a piano, and the sound is recorded on a microphone. We'll refer to the voltage signal produced by the microphone as $x(t)$ ([link]).

The time-domain plot of a piano recording does not give much insight into what was played.



The frequency domain representation of the piano recording shows us that three keys were played, a chord.

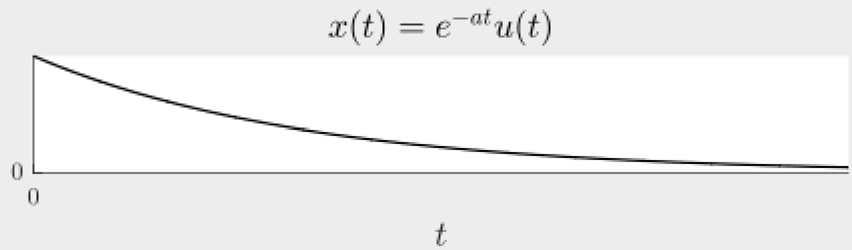


The time domain plot of the signal does not give us much of idea about what the signal is. But now let's take a look at $|X(j\omega)|$ the magnitude of the CTFT of $x(t)$ ([link]). There are three large spikes in the frequency domain, which makes sense because there were three keys depressed to make the chord. The CTFT has provided us helpful analysis: we have a better idea of what kind of signal $x(t)$ is by looking at its CTFT than by looking at its time-domain plot.

As indicated above, many (but not all) signals have a CTFT representation. Like with the CTFS, a few conditions on $x(t)$ are necessary if it is to be reproduced from its CTFT $X(j\omega)$: it must be absolutely integrable ($\int_{-\infty}^{\infty} |x(t)| dt < \infty$), and have a finite number of maxima/minima and discontinuities over any bounded interval.

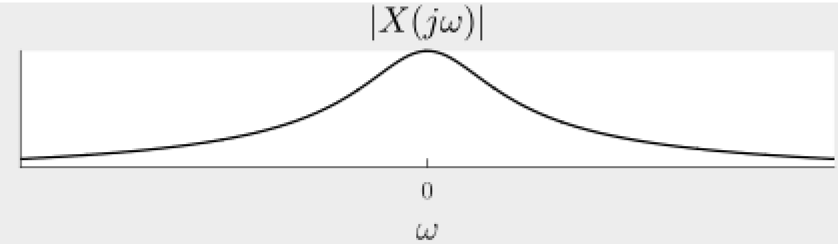
Example:
The CTFT of an Exponential Function

Now that we have defined the CTFT, let's try one out for an example. We'll start with the signal $x(t) = e^{-at}u(t)$, $a > 0$ ([link]) and find its CTFT $X(j\omega)$.

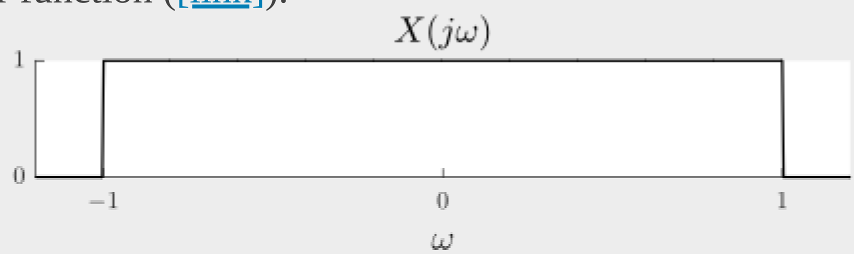


To find $X(j\omega)$, we simply put $x(t)$ into the CTFT formula:

$$\begin{aligned} X(j\omega) &= \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \\ &= \int_{0}^{\infty} e^{-at} e^{-j\omega t} dt \\ &= \int_{0}^{\infty} e^{-(a+j\omega)t} dt \\ &= -\frac{1}{a+j\omega}\left[e^{-(a+j\omega)t}\right]_0^{\infty} \\ &= -\frac{1}{a+j\omega}[0-1] \\ &= \frac{1}{a+j\omega} \end{aligned}$$

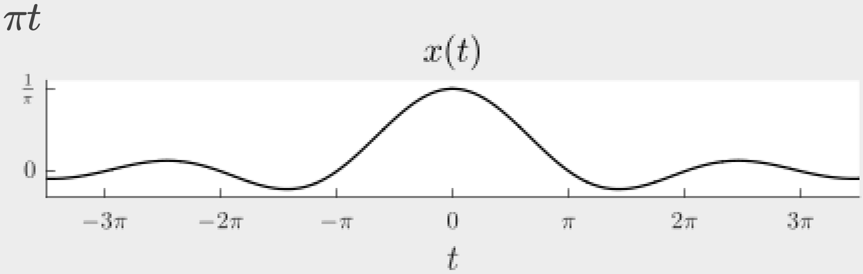


Example:
Finding the Inverse CTFT
Suppose now that we are told what a signal's CTFT is, and we would like to recover that signal from its CTFT. In order to do that, all we need to do is use the CTFT "synthesis formula." The CTFT we are given is a rectangular function ([link]).



To find $x(t)$, we will put the $X(j\omega)$ given into the inverse CTFT formula:

$$
\begin{aligned}
x(t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\omega)e^{j\omega t}d\omega \\
&= \frac{1}{2\pi}\int_{-1}^{1} 1e^{j\omega t}d\omega \\
&= \frac{1}{2\pi}\left[\frac{1}{jt}e^{j\omega t}\right]_{-1}^{1} \\
&= \frac{1}{2\pi}\left[\frac{1}{jt}e^{jt} - \frac{1}{jt}e^{-jt}\right] \\
&= \frac{1}{2\pi}\left[\frac{e^{jt} - e^{-jt}}{jt}\right] \\
&= \frac{1}{\pi t}\left[\frac{e^{jt} - e^{-jt}}{2j}\right] \\
&= \frac{\sin(t)}{\pi t}
\end{aligned}
$$



Continuous-Time Fourier Transform Properties

Having defined the CTFT, we will now examine its properties more closely. The first to consider is the **linearity** of the CTFT. Let signal $x(t)$ have a CTFT $X(j\omega)$ and signal $g(t)$ have a CTFT of $G(j\omega)$, with α and β as constants. Consider then the linear combination:
$\alpha x(t) + \beta g(t)$.
The CTFT of that combined signal is:

$$\begin{aligned}&\int_{-\infty}^{\infty} (\alpha x(t) + \beta g(t)) e^{-j\omega t} dt \\ =&\int_{-\infty}^{\infty} \left(\alpha x(t) e^{-j\omega t} + \beta g(t) e^{-j\omega t}\right) dt \\ =&\alpha \int_{-\infty}^{\infty} x(t) e^{-j\omega t} + \beta \int_{-\infty}^{\infty} g(t) e^{-j\omega t} dt \\ =&\alpha X(j\omega) + \beta G(j\omega)\end{aligned}.$$

The linearity of the CTFT is not much of a surprise, since it is essentially just an integral (and the integral operator is linear), but nonetheless it will be useful as we look at other properties.

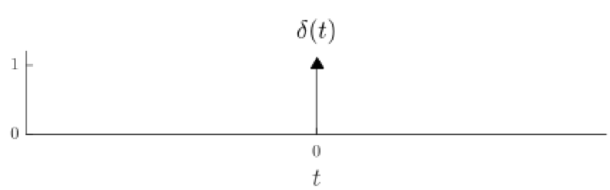
Common CTFT Pairs

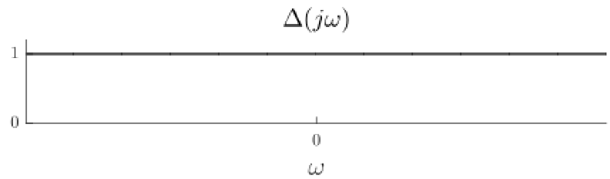
A CTFT "pair" is some signal--say, $x(t)$--and its corresponding CTFT representation $X(j\omega)$. We will refer to a CTFT pair using the following notation:
$x(t) \leftrightarrow X(j\omega)$
In this section we will detail some of the most common CTFT pairs.

Deltas and Sinusoids

Recall the dirac delta function $\delta(t)$ ([link]). $\delta(t)$ has the property that it is 0 at all $t \neq 0$, and that $\int_{-\infty}^{\infty} \delta(t) dt = 1$.
It is straightforward to find its CTFT:

$$\begin{aligned}\Delta(j\omega) &= \int_{-\infty}^{\infty} \delta(t) e^{-j\omega t} dt \\ &= \int_{-\infty}^{\infty} \delta(t) e^{-j\omega 0} dt \\ &= \int_{-\infty}^{\infty} \delta(t) 1 dt \\ &= 1\end{aligned}$$

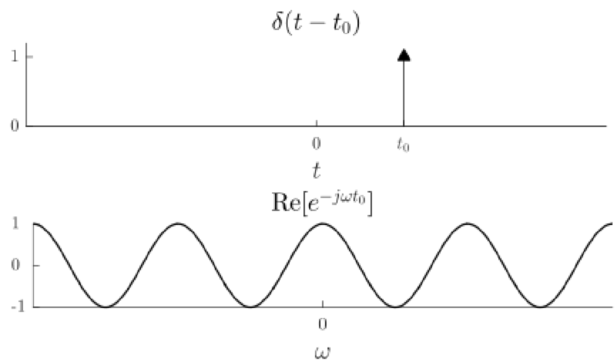




We'll now shift the delta function a bit ([link]) and see what happens.

$$\begin{aligned}\Delta_s(j\omega) &= \int_{-\infty}^{\infty} \delta(t-t_0)e^{-j\omega t}dt \\ &= \int_{-\infty}^{\infty} \delta(t-t_0)e^{-j\omega t_0}dt \\ &= e^{-j\omega t_0}\end{aligned}$$

A delta in time corresponds to a complex sinusoid in frequency, with the frequency of this sinusoid determined by how much the delta was shifted in the time domain.



We may wonder what happens the other way around. Suppose now we have $x(t)$ as a complex sinusoid in time. It is not immediately clear how to find its CTFT:

$\int_{-\infty}^{\infty} e^{j\omega_0 t}e^{-j\omega t}dt.$

The problem is that this integral is not defined, because the complex exponential goes on forever. What we can do is leverage the fact that the CTFT and inverse CTFT formulas are very similar, differing only by a scaling factor and negation in the exponent.

So, suppose $x(t)$ and $X(j\omega)$ are a CTFT pair:

$x(t) \leftrightarrow X(j\omega)$

What now might happen if we have a time-domain signal that looks like another signal's CTFT?

$X(t) \leftrightarrow\ ???$

Let's find this new signal's CTFT, using a temporary change of variables along the way:

$$\begin{aligned}
\text{CTFT}\ \{X(t)\} &= \int_{-\infty}^{\infty} X(t)e^{-j\omega t}dt \\
&\text{Let } \hat{\omega} = t, \hat{t} = \omega \\
&= \int_{-\infty}^{\infty} X(j\hat{\omega})e^{-j\hat{t}\hat{\omega}}d\hat{\omega} \\
&= 2\pi\frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\hat{\omega})e^{-j\hat{t}\hat{\omega}}d\hat{\omega} \\
&= 2\pi\left(\frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\hat{\omega})e^{j\hat{\omega}(-\hat{t})}d\hat{\omega}\right) \\
&= 2\pi x(-\hat{t}) \\
&\text{Change variables back} \\
&= 2\pi x(-j\omega)
\end{aligned}$$

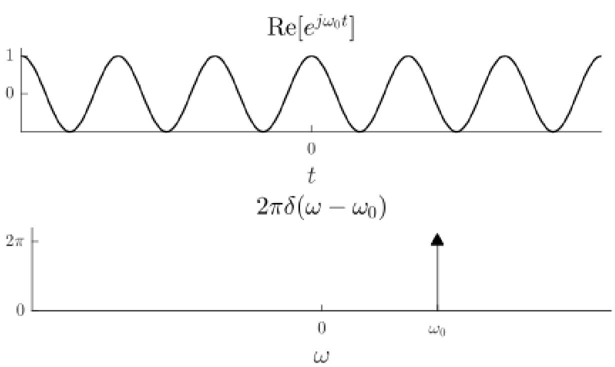
After we put in a temporary change of variables, we can see how much the CTFT of $X(t)$ resembles the inverse CTFT of $x(j\omega)$ (recall that the j in the expression $X(j\omega)$--or in this case, $x(j\omega)$--is simply a way to mark that we are speaking of a CTFT). It leads us to this result:

$X(t) \leftrightarrow 2\pi x(-j\omega)$

It also works the other way around:

$\frac{1}{2\pi}X(-t) \leftrightarrow x(j\omega)$

The whole point of that exercise was to figure out what the CTFT of a complex sinusoid is. We would like to know the CTFT of a signal like $e^{j\omega_0 t}$ ([link]).



However, it was not clear how to integrate that over $-\infty$ to ∞. The good news is that we know a signal whose CTFT was a complex sinusoid: a shifted delta.

$\delta(t-t_0) \leftrightarrow e^{-j\omega t_0}$

We'll change the shift direction so that the exponent in the CTFT is positive:

$\delta(t+t_0) \leftrightarrow e^{j\omega t_0}$

Now recall the relationship we derived above:

$x(t) \leftrightarrow X(j\omega)$

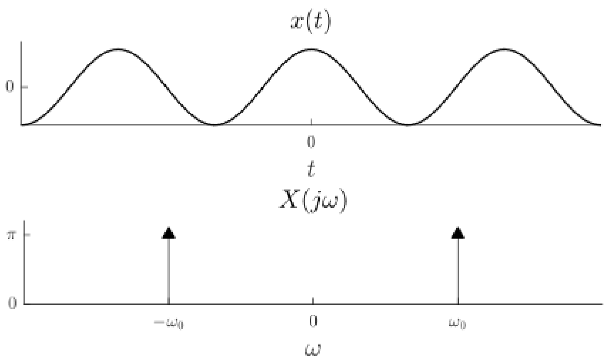
$X(t) \leftrightarrow 2\pi x(-j\omega)$

We can use that relationship to fill in the blanks here (remembering for the last step that the delta function is even):

$\delta(t+t_0) \leftrightarrow e^{j\omega t_0}$

$e^{j\omega_0 t} \leftrightarrow\ ??? = 2\pi\delta(-\omega+\omega_0) = 2\pi\delta(\omega-\omega_0)$

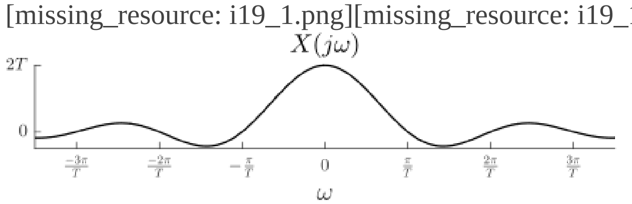
Having done this, we can then use the linearity property of the CTFT to determine the CTFT for a signal $x(t) = \cos(\omega_0 t)$ ([link]). Since $\cos(\omega_0 t) = \frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2}$, then we have $X(j\omega) = \pi\delta(\omega - \omega_0) + \pi\delta(\omega + \omega_0)$. And, of course, the same logic would apply to a sine as well:
$\sin(\omega_0 t) = \frac{e^{j\omega_0 t} - e^{-j\omega_0 t}}{2j} \leftrightarrow \frac{\pi}{j}\delta(\omega - \omega_0) - \frac{\pi}{j}\delta(\omega + \omega_0)$
So based upon the above, we can conclude that it is a general CTFT property that a sinusoid in one domain corresponds to deltas in the other.



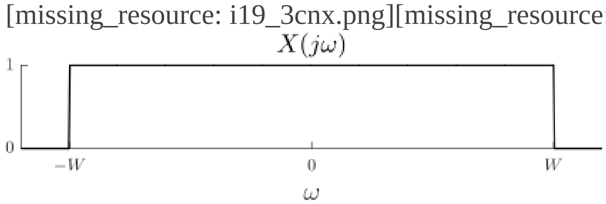
Sincs and Rectangles

Another common CTFT pair has to do with sinc and rectangle functions. If $x(t)$ is a rectangle function, then its CTFT $X(j\omega)$ is a sinc ([link]):
$x(t) = \begin{cases} 1 & |t| \leq T \\ 0 & |t| > T \end{cases} \leftrightarrow \frac{2\sin(T\omega)}{\omega}$
Likewise, if it is $X(j\omega)$ that is a rectangle, then $x(t)$ will be a sinc ([link]):
$\frac{\sin(Wt)}{\pi t} \leftrightarrow X(j\omega) = \begin{cases} 1 & |\omega| \leq W \\ 0 & |\omega| > W \end{cases}$
Just as with deltas and sinusoids, rectangles in one CTFT domain correspond with sinc functions in the other.

[missing_resource: i19_1.png][missing_resource: i19_1.eps]



[missing_resource: i19_3cnx.png][missing_resource: i19_3cnx.eps]



[missing_resource: i19_4cnx.eps]

Summary of CTFT Pairs
[link] summarizes the CTFT pairs we have covered, along with a couple other ones, as well. It can often be useful to look up a CTFT in a table like this (rather than working out the CTFT integral each time).

$x(t)$	$X(j\omega)$
$\delta(t)$	1
1	$2\pi\delta(\omega)$
$e^{j\omega_0 t}$	$2\pi\delta(\omega-\omega_0)$
$\cos(\omega_0 t)$	$\pi\delta(\omega-\omega_0)+\pi\delta(\omega+\omega_0)$
$\sin(\omega_0 t)$	$\frac{\pi}{j}\delta(\omega-\omega_0)-\frac{\pi}{j}\delta(\omega+\omega_0)$
$u(t)$	$\pi\delta(\omega)+\frac{1}{j\omega}$
$e^{-at}u(t)\ ,\ a>0$	$\frac{1}{a+j\omega}$
$x(t)=\begin{cases}1 & -T<t<T\\ 0 & \text{else}\end{cases}$	$\frac{2\sin(\omega T)}{\omega}$
$\frac{\sin(Wt)}{\pi t}$	$X(j\omega)=\begin{cases}1 & -W<\omega<W\\ 0 & \text{else}\end{cases}$

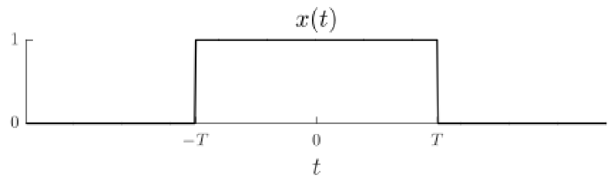
CTFT Time/Frequency Relationships

For a given CTFT pair, making a change to one side of the pair will of course influence the other (because a signal and its CTFT have a one-to-one relationship). We'll now take a look at some of the ways a change in one domain corresponds to a change in the other.

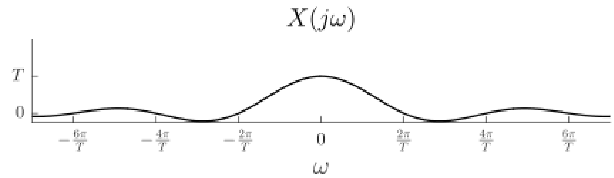
Time/Frequency Scaling

Recall that a rectangle and sinc are a CTFT pair ([link]). [link] shows what happens when we double the time scale for $x(t)$. Notice how it halves the frequency scale for $X(j\omega)$. Then, as you might expect, [link] shows what when we halve the time scale for $x(t)$, it doubles the frequency scale for $X(j\omega)$.

A rectangle and sinc are CTFT pairs.

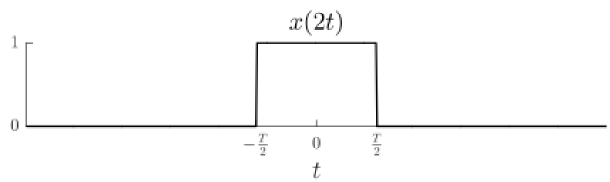


[missing_resource: i28_1.eps]

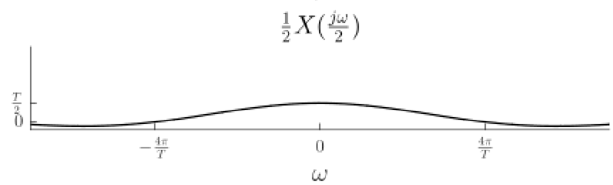


[missing_resource: i28_2.eps]

Doubling the time scale of a signal halves the frequency scale of its CTFT.

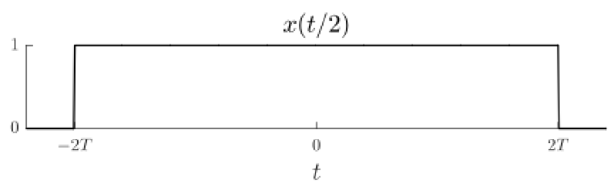


[missing_resource: i28_3.eps]

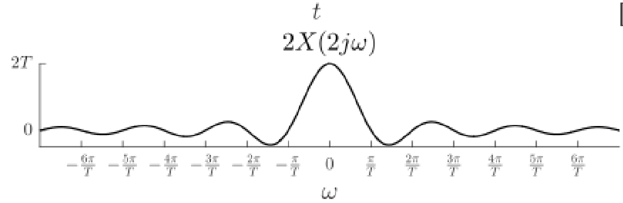


[missing_resource: i28_4.eps]

Halving the time scale of a signal doubles the frequency scale of its CTFT.



[missing_resource: i28_5.eps]



[missing_resource: i28_6.eps]

For a more general mathematical expression of what's going on here, let $x(t)$ and $X(j\omega)$ be a CTFT pair $x(t) \leftrightarrow X(j\omega)$. If we scale the time variable of $x(t)$ by some constant scalar value a , it will scale the frequency variable of the CTFT by $\frac{1}{a}$:

$x(at) \leftrightarrow \frac{1}{|a|}X(j\frac{\omega}{a})$.

Notice that the magnitude of the CTFT is also scaled. Derivation of this property needs only a simple change of variables:

$$\mathrm{CTFT}\{x(at)\} = \int_{-\infty}^{\infty} x(at)e^{-j\omega t}dt$$

$$\text{Let } u = at \;\rightarrow\; t = \frac{u}{a} \;,\; du = adt$$

$$= \int_{-\infty}^{\infty} x(u)e^{-j\omega\frac{u}{a}}\frac{1}{|a|}du$$

$$= \frac{1}{|a|}\int_{-\infty}^{\infty} x(u)e^{-j\frac{\omega}{a}u}du$$

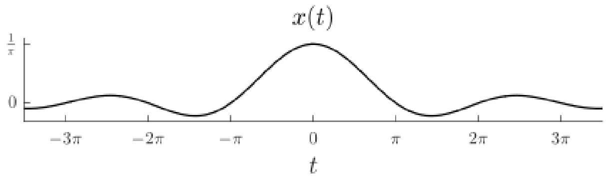
$$= \frac{1}{|a|}X(j\frac{\omega}{a})$$

(The reason for the magnitude bars around the a is because if it were negative, the integral would run from ∞ to $-\infty$, in which case we would negate it to go back from $-\infty$ to ∞).

As you can see from the figures of the rectangle and its CTFT sinc, it is not possible for a signal to be simultaneously narrow in both time and frequency. If it is very narrow in one domain, it will be spread out in the other. Nor is it possible, therefore, for a signal to be spread out in both domains. This is sometimes referred to as the time/frequency **uncertainty principle**. Much like the principle in physics that states that a particle's momentum and position cannot be simultaneously determined (or localized), so also a signal cannot be localized in both time and frequency.

Time/Frequency Shifting

Another way a signal can be modified in the time domain is by shifting it. Recall that if $x(t)$ is a sinc, then $X(j\omega)$ is a rectangle ([link]). [link] shows what happens to the CTFT if we shift $x(t)$ over to the right a bit.



[missing_resource: i32_1.eps]
[missing_resource: i32_2.png][missing_resource: i32_2.eps]

Shifting a signal in time modulates its CTFT (multiplies it by a complex sinusoid).

[missing_resource: i32_3.png][missing_resource: i32_3.eps]
[missing_resource: i32_4.png][missing_resource: i32_4.eps]

When $x(t)$ is shifted in time, the CTFT is multiplied by a complex sinusoid. Another word for this multiplication by a sinusoid is **modulation**:
$x(t-t_0) \leftrightarrow e^{-j\omega t_0}X(j\omega)$
Due to the duality of the CTFT, the principle also applies in the other direction. Modulation in time results in a shift in frequency:
$e^{j\omega_0 t}x(t) \leftrightarrow X(j(\omega-\omega_0))$

Convolution/Multiplication

One final time/frequency relationship is especially relevant in the study of signals and systems. Recall that for an LTI system with impulse response $h(t)$, the output $y(t)$ for a given input $x(t)$ can be found through convolution:

$y(t) = x(t) * h(t)$

This convolution occurs in the time domain. It is natural to wonder what the corresponding relationship between the input, output, and impulse response is in the frequency domain. To do that, we will look at the CTFT of $y(t)$, and simplify it using the convolution integral (note how we use the time shifting CTFT property along the way):

$$\begin{aligned} Y(j\omega) &= \int_{-\infty}^{\infty} y(t)e^{-j\omega t}\,dt \\ &= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} x(\tau)h(t-\tau)\,d\tau\right]e^{-j\omega t}\,dt \\ &= \int_{-\infty}^{\infty} x(\tau)\left[\int_{-\infty}^{\infty} h(t-\tau)e^{-j\omega t}\,dt\right]d\tau \\ &= \int_{-\infty}^{\infty} x(\tau)\left[CTFT\{h(t-\tau)\}\right]d\tau \\ &= \int_{-\infty}^{\infty} x(\tau)\left[e^{-j\omega\tau}H(j\omega)\right]d\tau \\ &= H(j\omega)\int_{-\infty}^{\infty} x(\tau)e^{-j\omega\tau}\,d\tau \\ &= H(j\omega)X(j\omega) \end{aligned}$$

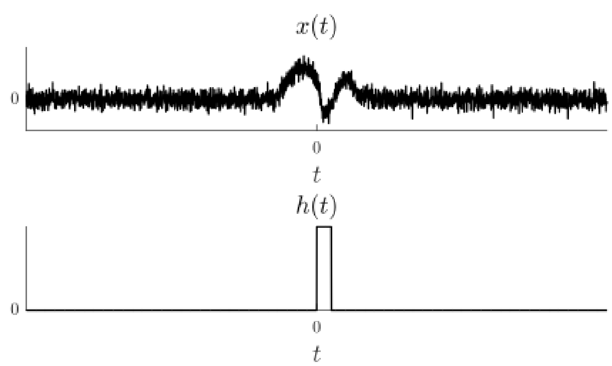
So we now see another CTFT relationship. Convolution in time corresponds to multiplication in frequency:

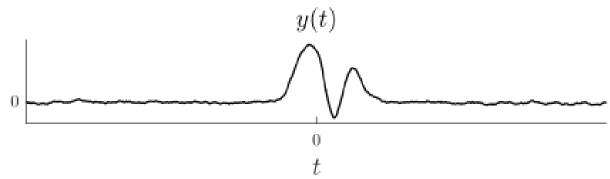
$h(t) * x(t) \leftrightarrow H(j\omega)X(j\omega)$

We also can better understand why $H(j\omega)$, the CTFT of $h(t)$, is called the system's **frequency response**. It tells us how the system scales various frequency components. The relationship $Y(j\omega) = H(j\omega)X(j\omega)$ indicates that each frequency component in $X(j\omega)$ is scaled by $H(j\omega)$ to produce the output frequency component $Y(j\omega)$.

This property can give us additional insight in to how systems modify input signals. For example, we have already seen that a system with a pulse for an impulse response can take signals with high frequency noise and denoise them by "averaging" or "blurring" out the noise ([link]).

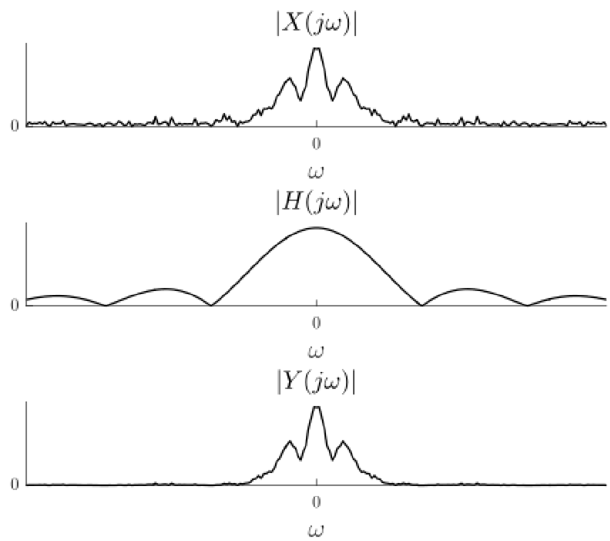
Convolving a signal that has high frequency noise with a pulse results in the signal being de-noised.





Looking at that same operation in the frequency domain gives another picture of how the de-noising works ([link]). A signal with high frequency noise will have a considerable amount of high-frequency components. The frequency response of the system is close to 0 at high frequencies, while being more or less close to 1 at low frequencies. Because the CTFT of the output is the product of the CTFT of the input and the frequency response, the result is for the system to "zero out" the high-frequency noise components. This also shows us why the de-noising operation isn't perfect: the frequency response is not exactly 1 for low frequencies, nor is it exactly 0 for higher frequencies.

The frequency response of a de-noising system has values close to 0 for high frequency, thus "zeroing out" those frequencies when multiplied by the CTFT of the input.



Summary of Relationships
We have seen several ways in which modifications in one domain of a CTFT pair result in certain changes in the other. These properties, as well as one on differentiation, are summarized in [link].

Property	$x(t)$	$X(j\omega)$
Linearity	$\alpha x(t) + \beta g(t)$	$\alpha X(j\omega) + \beta G(j\omega)$
Time scaling	$x(at)$	$\frac{1}{\lvert a\rvert} X(j\frac{\omega}{a})$

Time shifting	$x(t-t_0)$	$e^{-j\omega t_0}X(j\omega)$
Modulation	$x(t)e^{j\omega_0 t}$	$X(j(\omega-\omega_0))$
Time convolution	$x(t)*h(t)$	$X(j\omega)H(j\omega)$
Frequency convolution	$x(t)g(t)$	$\frac{1}{2\pi}X(j\omega)*G(j\omega)$
Time differentiation	$\frac{d^n}{dt^n}x(t)$	$(j\omega)^n X(j\omega)$

CTFT Preservation of Inner Product and Energy

Whereas most of the time/frequency properties we have considered have a reciprocal relationship, the final property is more of an identity. Suppose we have two continuous-time infinite length signals, $x(t)$ and $g(t)$, with corresponding CTFTs $X(j\omega)$ and $G(j\omega)$. The inner product of the two signals is simply a scaled version of the inner product of their CTFTs (and vice versa):

$$\langle x(t), g(t)\rangle = \frac{1}{2\pi}\langle X(j\omega), G(j\omega)\rangle$$

$$\int_{-\infty}^{\infty} x(t)g^*(t)dt = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\omega)G^*(j\omega)d\omega$$

This property is known as **Parseval's Theorem**, and it is pretty remarkable. The two signals look very different from their CTFTs, but their inner products are the same, simply scaled by $\frac{1}{2\pi}$. This property might come in handy in cases where finding the inner product might be easier in one domain than the other.

It can also be used in another way, by making the two signals identical. In this case then we have:

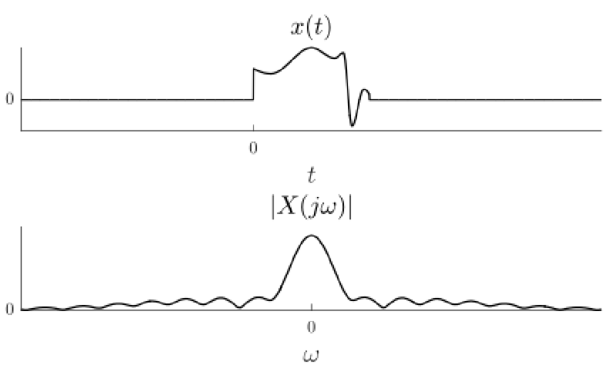
$$\langle x(t), x(t)\rangle = \frac{1}{2\pi}\langle X(j\omega), X(j\omega)\rangle$$

$$\| x(t) \|_2^2 = \frac{1}{2\pi} \| X(j\omega) \|_2^2$$

This shows that the CTFT preserves the energy of a signal (with a $\frac{1}{2\pi}$ scaling factor), and is known as **Plancherel's Theorem**.

CTFT Symmetry

Most of the CTFTs we have seen exhibit a kind of symmetry. Consider the $x(t)$ and $|X(j\omega)|$ in [link]. $|X(j\omega)|$ is even, being symmetric about $\omega = 0$, even though $x(t)$ did not have such symmetry.



There are two reasons this is the case. First, it is because the complex sinusoids that go in to the CTFT formula have certain symmetries, namely, that they are **conjugate symmetric**. If some $s(t)$ is conjugate symmetric, that means $s^*(t) = s(-t)$ (or, equivalently, $s(t) = s^*(-t)$). And second, it is because $x(t)$ is real-valued. When we combine complex sinusoid conjugate symmetry with a real-valued signal in the CTFT, then we will see that the CTFT is also conjugate symmetric:

$$\begin{aligned} X^*(j\omega) &= \left(\int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt\right)^* \\ &= \int_{-\infty}^{\infty} \left(x(t)e^{-j\omega t}\right)^* dt \\ &= \int_{-\infty}^{\infty} (x(t))^* \left(e^{-j\omega t}\right)^* dt \\ &= \int_{-\infty}^{\infty} x(t)\left(e^{-j\omega t}\right)^* dt \quad (\text{as } x(t) \text{is real}) \\ &= \int_{-\infty}^{\infty} x(t)e^{-(-j)\omega t}dt \\ &= \int_{-\infty}^{\infty} x(t)e^{-j(-\omega)t}dt \\ &= X(-j\omega) \end{aligned}$$

Conjugate symmetry implies many other symmetries. For example, if $s(t)$ is conjugate symmetric, then its real part and its magnitude are even, while its imaginary part and its phase are odd.

So if a signal is real, then its CTFT is conjugate symmetric. If the signal itself also has symmetry by being either even or odd, then its CTFT will not only be conjugate symmetric, it will be even (for even time-domain signals) or odd (for odd time-domain signals). All of these symmetry properties are summarized in [link].

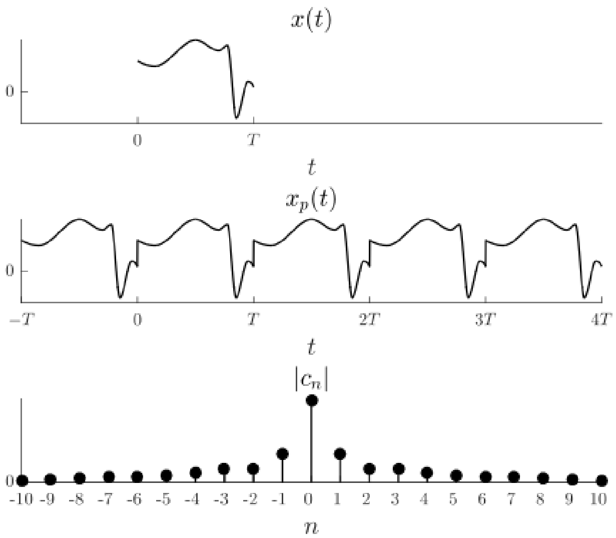
$x(t)$	$X(j\omega)$	Re $(X(j\omega))$	Im $(X(j\omega))$	$\lvert X(j\omega)\rvert$	$\angle X(j\omega)$
real	conjugate symmetric: $X(-j\omega) = X(j\omega)^*$	even	odd	even	odd
real and even	real and even	even	zero	even	$0, \pi$
real and odd	imaginary and odd	zero	odd	even	$\pm\frac{\pi}{2}$
imaginary	conjugate odd symmetric $X(-j\omega) = -X(j\omega)^*$	odd	even	even	odd
imaginary and even	imaginary and even	zero	even	even	$\pm\frac{\pi}{2}$
imaginary	real and odd	odd	zero	even	$0, \pi$

and odd					

CTFT/CTFS Relationship

Recall that we developed the CTFT from the CTFS. The CTFS is a frequency domain representation for finite-length or periodic signals.

Finite-length signals ($x(t)$ above) or periodic signals ($x_p(t)$) can be represented as a weighted sum of complex sinusoids $c_n e^{j\frac{2\pi}{T}nt}$.



Also recall the formula for a signal's representation with CTFS coefficients, letting $\omega_0 = \frac{2\pi}{T}$:

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\omega_0 nt}$$

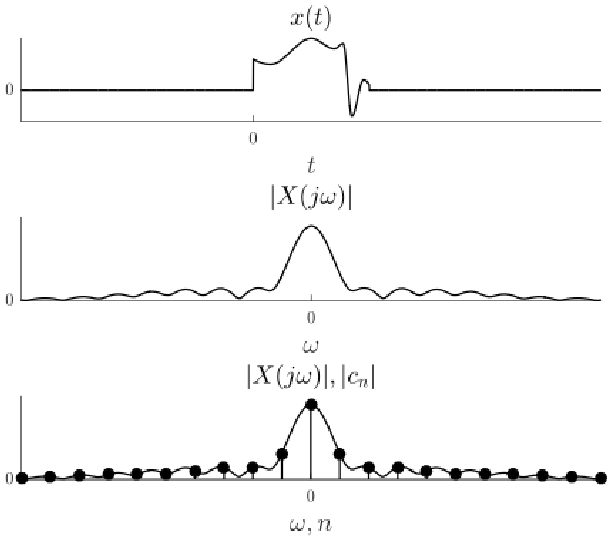
$$c_n = \frac{1}{T}\int_0^T x(t)e^{-j\omega_0 nt}dt$$

The formula for c_n can actually be expressed as a CTFT, though only at certain points:

$$\begin{aligned} c_n &= \frac{1}{T}\int_0^T x(t)e^{-j\omega_0 nt}dt \\ &= \left[\frac{1}{T}\int_0^T x(t)e^{-j\omega t}dt\right]_{\omega=n\omega_0} \\ &= \frac{1}{T}X(jn\omega_0) \end{aligned}$$

So, then, as shown in [link], a CTFS of a finite-length signal is actually just a sampled version of a CTFT for the same signal (though zero-padded to be infinite length). Every c_n is simply the value of the CTFT at location $n\omega_0$.

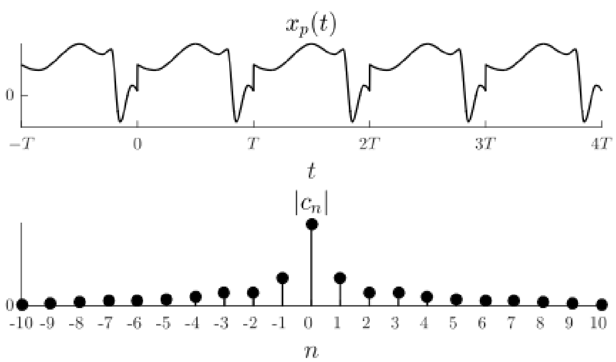
When a finite-length signal is made to be infinite-length by adding zeros before and after it, we can see that the CTFS of the finite-length signal is simply a sampled version of the CTFT of the infinite-length signal.



So we know what happens when we do a CTFT on a finite-length signal by considering the expanded time-domain portion of it as being zero: the CTFS of the finite-length signal is a sampled version of the CTFT of the infinite-length signal.

Of course, the CTFS applies to periodic signals as well. The CTFT is used for infinite-length signals, and a periodic signal is also infinite in length (it repeats forever!), so we can also consider what happens if we take the CTFT of the periodic signal. Recall that the CTFS of the a periodic signal $x_p(t)$ is identical to the CTFS of the finite-length signal $x(t)$ that is a single period of $x_p(t)$ ([link]).

A periodic signal and its CTFS coefficients.

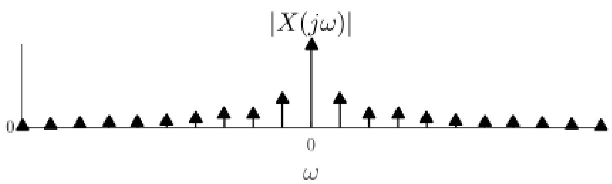


The CTFT of a periodic signal is:

$$\begin{aligned} X(j\omega) &= \int_{-\infty}^{\infty} x_p(t) e^{-j\omega t} dt \\ &= \int_{-\infty}^{\infty} \left(\sum_{n=-\infty}^{\infty} c_n e^{j\omega_0 n t} \right) e^{-j\omega t} dt \\ &= \sum_{n=-\infty}^{\infty} c_n \left(\int_{-\infty}^{\infty} e^{j\omega_0 n t} e^{-j\omega t} dt \right) \\ &= \sum_{n=-\infty}^{\infty} c_n \left(\mathrm{CTFT}\{e^{j\omega_0 n t}\} \right) \\ &= \sum_{n=-\infty}^{\infty} c_n \left(2\pi\delta(\omega - \omega_0 n) \right) \\ &= 2\pi \sum_{n=-\infty}^{\infty} c_n \delta(\omega - \omega_0 n) \end{aligned}$$

So the CTFT of a periodic signal is a series of delta functions, each scaled by its corresponding CTFS coefficients ([link]).

The CTFT of a periodic signal is a series of delta functions at $\omega_0 n$, the scales of which are the same as the signal's CTFS coefficients.



This representation makes sense, because we already knew (from the CTFS) that periodic signals can be composed of a sum of exponential signals. The CTFT gives a frequency representation of an exponential as a delta function, so it follows that the CTFT of a periodic signal would be a series of delta functions.

CTFT Properties in Action: Amplitude Modulation

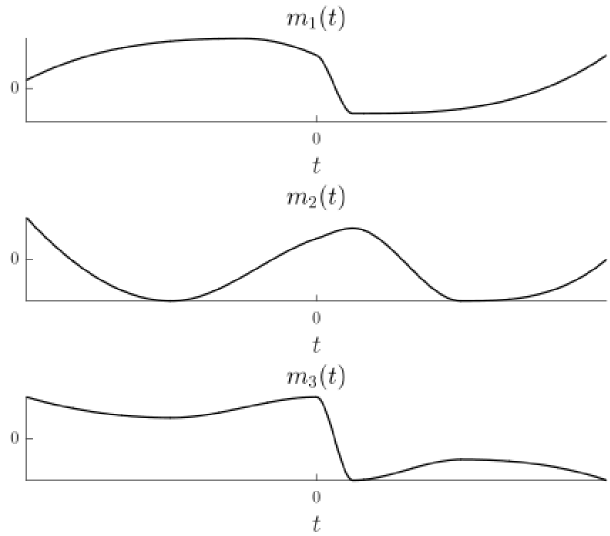
Having reviewed the most important properties of the CTFT, we can now put some of them to use in understanding the **amplitude modulation** operation.

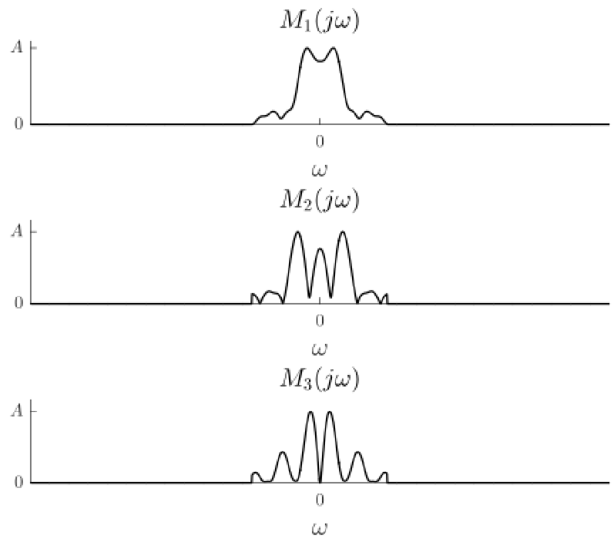
Have you ever wondered how all of the different radio, television, satellite, and cell phone signals around you don't get hopelessly mixed up? Football games, soap operas, Tejano music, hundreds of cell phone conversations...all travel through the same airspace, yet we expect our TV/radio/phone to isolate only the one particular information stream that we're interested in.

One of the ways many different signals can be transmitted through the same space (and recovered individually) is through amplitude modulation. As you might have guessed, this is the way AM radio handles the task. We can better understand how and why it works with the help of the CTFT.

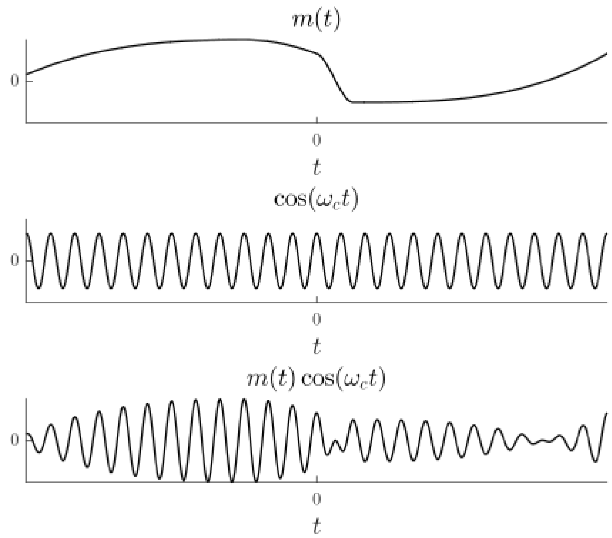
Suppose there are three signals we would like to transmit across the same airspace, and at the same time, signals m_1, m_2, and m_3 ([link]). Now, if we simply added them together and transmitted the sum, we wouldn't be able to separate them again (if you are told the sum of three numbers is 20, there is no way

from that information alone to know what the three numbers were). Of course, it is possible to send each one after the other, but right now we'd like to see if there is a way to send them simultaneously. So we'll need to do something else. Lets first look at their CTFTs ([link]).

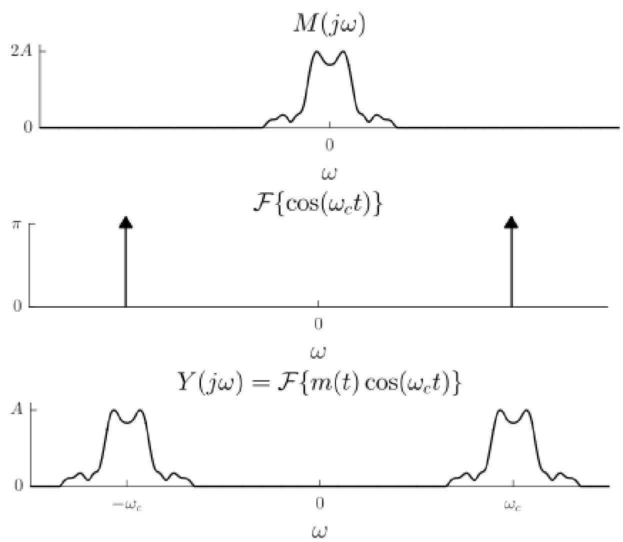


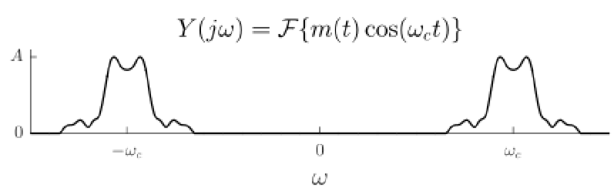


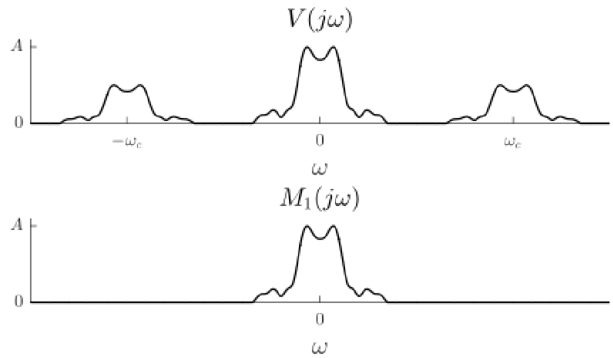
We can't simply just add them together (due to linearity, that is the same as adding them in the time domain!). But we can see that there is a lot of empty space in the frequency domain. We could shift them in the frequency domain and still be able to send them simultaneously. The modulation property we considered earlier states that we can shift a signal in its frequency domain by multiplying it by a sinusoidal in the time domain ([link]).



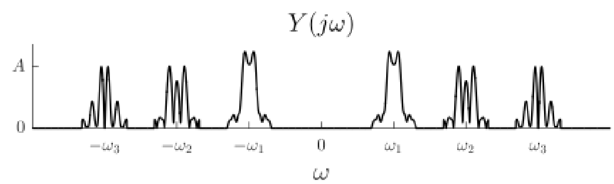
We will recall that multiplication in the time domain corresponds to convolution in the frequency domain. Combining that property with the fact that a cosine in time corresponds to two deltas in frequency, we can see what this modulation looks like in the frequency domain ([link]). Of course, we are going do need to eventually undo this operation. In order to do that, all we have to do is multiply it by the same cosine again, and then filter out the center section ([link]).

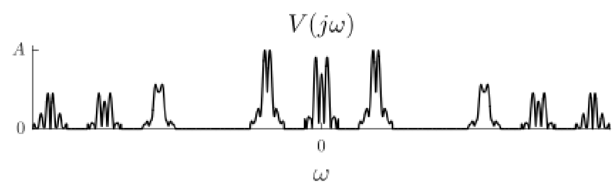




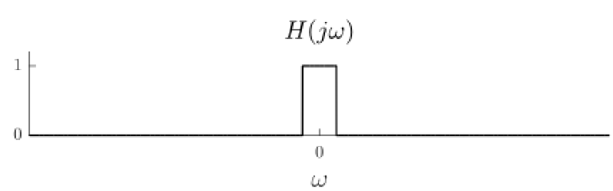


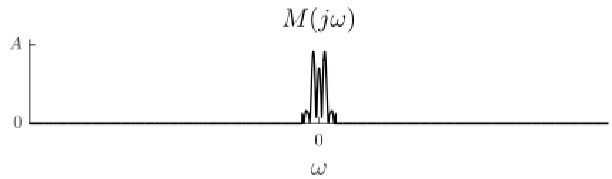
So then, if we modulate each of the three signals by a different frequency ([link]), then we can separate out the three signals in the frequency domain. To pull out a particular signal, say, the one modulated by ω_2, we will first "tune" in to it by multiplying by the carrier cosine again ([link]).





Right there in the center of the frequency spectrum, at $\omega = 0$, is the CTFT of the signal we want to extract. Of course, there is also quite a bit of junk on either side of it. So the next step is to apply a lowpass filter; this means convolving the signal with another signal whose CTFT looks like the one in [link]. Since convolution in time equals multiplication in the frequency domain, the final effect is to isolate just the CTFT of the desired signal ([link]).





So we have successfully recovered the signal of interest! To recover the other ones, we simply would appropriately "tune" in to their respective carrier frequencies. Those frequencies are the station numbers for various radio stations. The station 750 AM has a carrier frequency of 750 kHz.

Now, it should be noted that this is not *exactly* how amplitude modulation works. What was described above is called "double sideband suppressed carrier" modulation. Amplitude modulation, such as is used for AM radio, actually does something just slightly different. Rather than multiply $m(t)$ by the carrier sinusoid, $(1+m(t))$ is multiplied by the carrier sinusoid. The primary upside to this approach is that it allows for simpler retrieval, but it works in basically the same way.

In either case, this way of transmitting multiple signals across the same communications medium is an example of a "multiple access" scheme. There are a few different approaches to this problem of several different signals needing to be communicated across the same communication medium (e.g., telephone cables, over the air as a radio wave). As stated earlier, signals could be sent one after the other at different points in time. This is known as time division multiple access (TDMA). In practice, it amounts to users of the channel having access to all of it, but only at defined times. In our illustration of amplitude modulation, rather than divide the communication medium usage by time, we divided it by frequency allocation; this is an example of "frequency division multiple access" (FDMA). FDMA divides up a communication medium into a number of frequency "channels." This limits the amount of frequency bandwidth that each operator can use (lest the signals interfere with each other), but the benefit is they can use the channel at any time. A third scheme is "code division multiple access," (CDMA) in which instead of multiplying a given signal by a sinusoid, it is multiplied by a pseudo-random code.

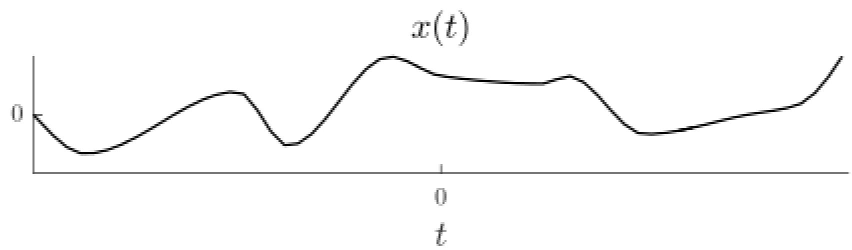
Windowing Infinite-Length Continuous Time Signals

The Frequency Domain Effects of Windowing

We have seen that the CTFT allows us to analyze signals of potentially infinite length (and finite-length signals as well, with the assumption they have a value of 0 outside the defined "finite" range). But there are many circumstances in which we might want to only consider a portion of a larger signal. This raises the question of how we are to mathematically "consider" only a portion, and what effect that has in the frequency domain.

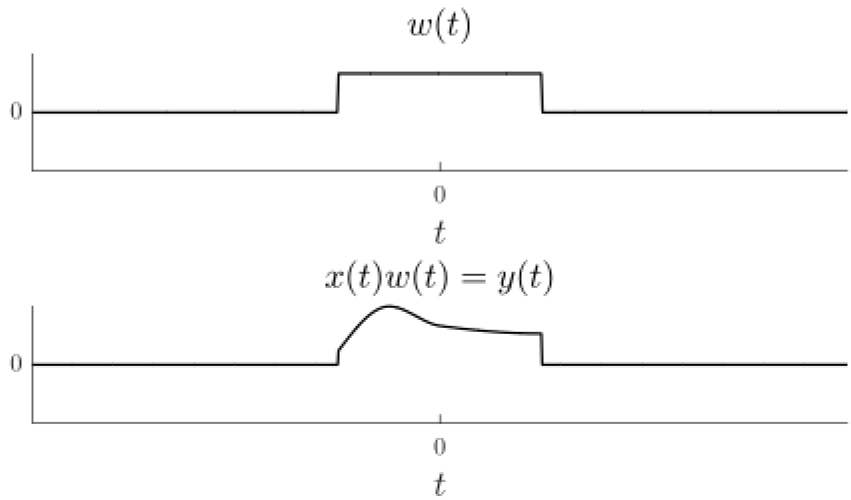
For example, suppose we have a lengthy signal $x(t)$, pictured in [link].

An infinite length signal, from which we would like to consider only a part.



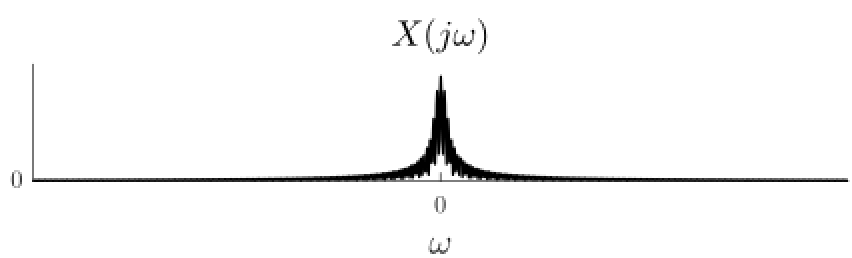
We'll assume that either it continues on forever (or is simply 0) outside the portion shown. But suppose we want to take the CTFT of just the center portion of it. We could mathematically "extract" that portion of the signal by multiplying $x(t)$ by another signal $w(t)$ ([link]).

Multiplying $x(t)$ by $w(t)$ extracts a portion of it for CTFT analysis.



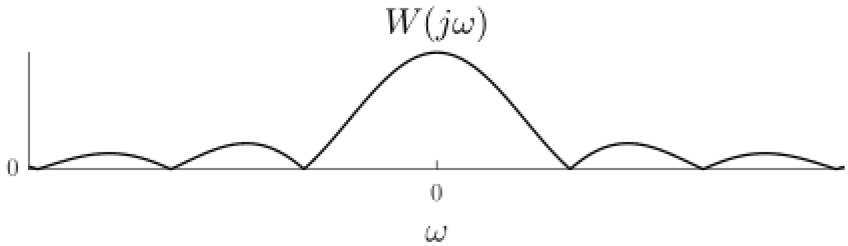
Having done that, it is straightforward to compute its CTFT. But we may wonder how the CTFT of the extracted portion compares to the CTFT of the original signal. To do this, we recall that multiplication in the time domain corresponds to convolution in the frequency domain. That means that $y(t) = x(t)w(t)$ corresponds to $Y(j\omega) = X(j\omega) * W(j\omega)$. So lets first look at $X(j\omega)$, [link], the CTFT of the original signal (the scale of the following CTFTs has been normalized to help with visualization).

The CTFT of the original infinite length signal $x(t)$.

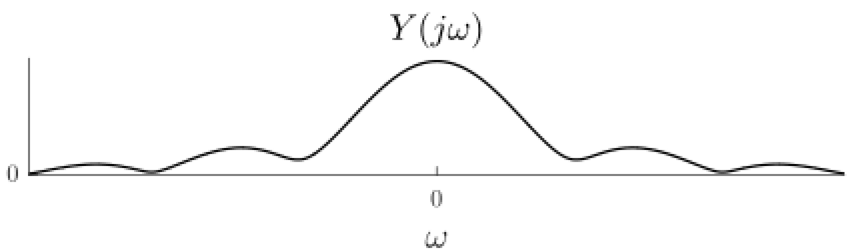


Since $x(t)$ was multiplied by $w(t)$ in time, $X(j\omega)$ will therefore be convolved with $W(j\omega)$ ([link]). The result is the extracted signal's CTFT, $Y(j\omega)$ ([link]).

The CTFT of the window $w(t)$.



The CTFT of $y(t) = x(t)w(t)$.



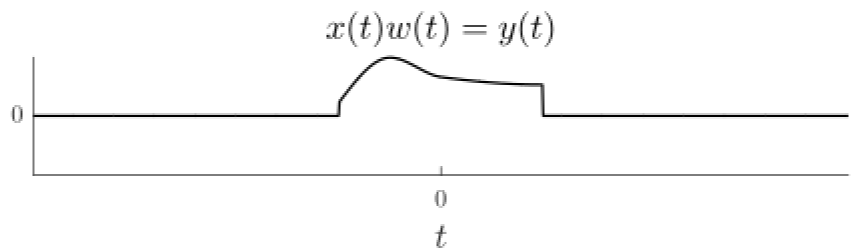
This analysis in the frequency domain helps us to better understand the nature of windowing, in two ways. First, we can see how the size of the window makes a difference. The wider the window, the more that the windowed signal will be like the original. Recalling the time/frequency CTFT relationship, we know that the more spread out the rectangular window is in time, the narrower the sinc will be in frequency. The narrower the window in frequency, the less it will "smear" the original CTFT as it produces the extracted signal's CTFT. In the extreme case, if the window is infinitely wide (i.e., $w(t) = 1$), then $y(t) = x(t)$. In the frequency domain, that $w(t)$ corresponds to a delta, and of course convolving $X(j\omega)$ with a delta will simply return $X(j\omega)$ again, thus also $Y(j\omega) = X(j\omega)$.

This smearing in the frequency domain illustrates what we know intuitively in the time domain, that the windowing operation results in lost information (so it is not reversible). It is impossible to recover $x(t)$ from the windowed version $y(t)$, and this loss of information in the time domain corresponds to the smearing/blurring/averaging in the frequency domain, which of course

is also a loss of information (if you are told only the average of two numbers, there is no way to recover what those two numbers were).

And the second way this CTFT analysis helps us to understand windowing is in showing how the type of window we use--not just its width--also has a significant effect. It might seem at first that the rectangular windowing operation above is relatively unbiased and neutral, objectively giving us a very good representation of the frequency content of the original signal at the time it was windowed. But as we saw in its CTFT, the windowed signal has a significantly higher proportion of higher frequency components. And look again at $y(t)$ ([link]). Note how it suddenly rises from 0 and then especially how it sharply drops again to 0 at the edges where it was widowed. *Such sudden changes were never present in the original signal* $x(t)$ *-- the windowing operation introduced them!* We can see this change in the frequency domain in the high frequency ripples in $W(j\omega)$ that by extension also appear $Y(j\omega)$.

While it might seem that a rectangular window $w(t)$ selects a portion of $x(t)$ in an unbiased way, it does introduce high frequencies: note how $y(t)$ jumps up from 0 and then falls back again sharply at the window edges.



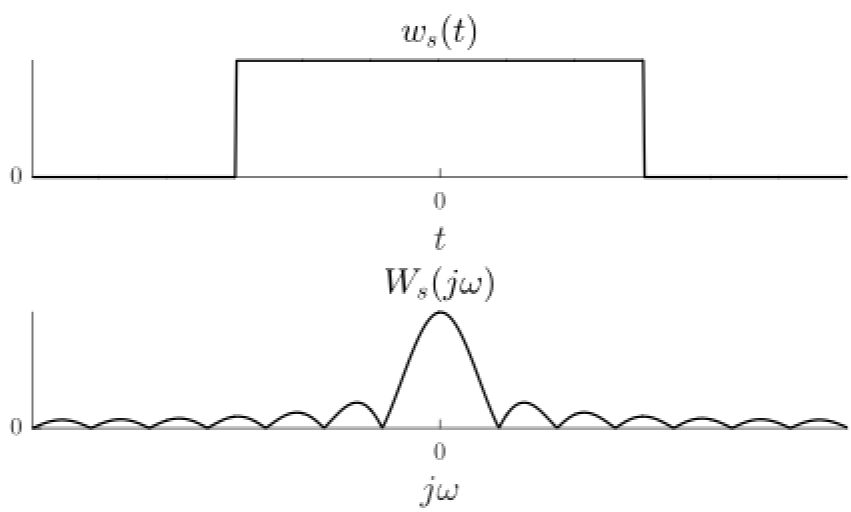
Types of Windows

So when it comes to the frequency domain (that is, $W(j\omega)$), there are two desirable properties for windows. First, we would like them to be narrow, which results in less "smearing" and lost information. And second, we

would prefer if they did not have ripples, which artificially introduce high frequencies in the windowed signal.

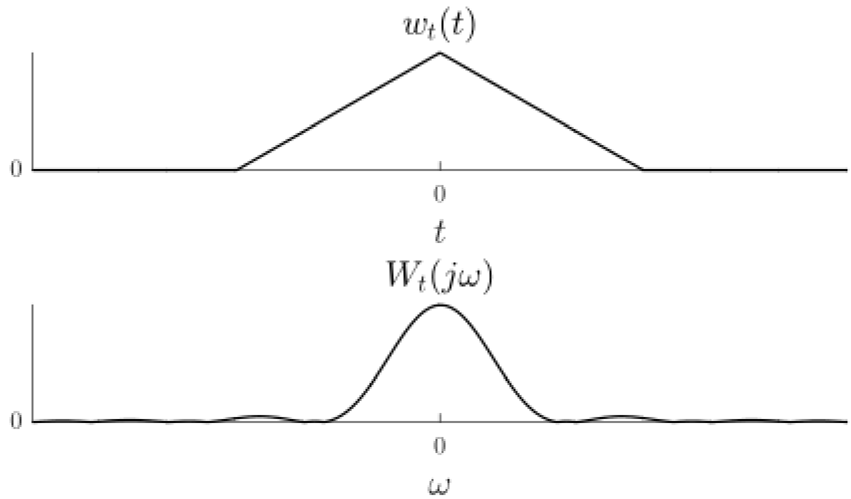
Unfortunately, for a given window length (in the time domain) it is not possible to achieve both goals. Some windows will be narrow but have ripples, while others will be wider and not have ripples. The square (or "rectangular") window is an example of the first type. [link] shows how it looks in the time and frequency domains.

The time and frequency domain plots of a rectangular window function.

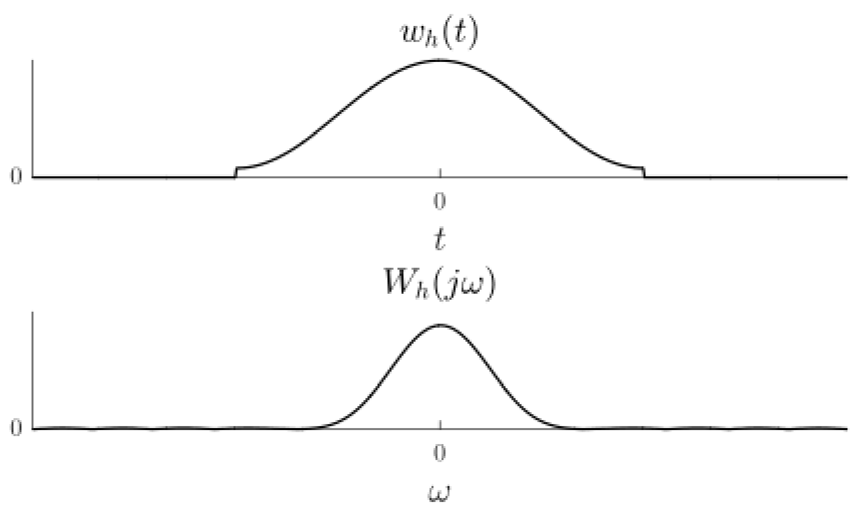


Two other windows are more of the second type. They both have the same width in the time domain as the rectangular window. But compared to it, they will not have (as much) rippling in the frequency domain, at the expense of being wider in the frequency domain. One is a triangle (or "Bartlett") window ([link]), and another similar one is the "Hamming" window ([link]).

The time and frequency domain plots of a triangular, or "Bartlett," window function.



The time and frequency domain plots of a Hamming window function.

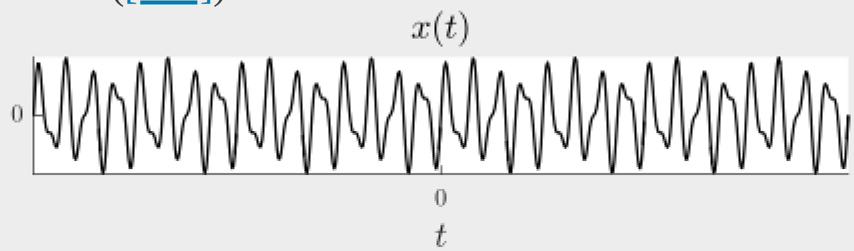


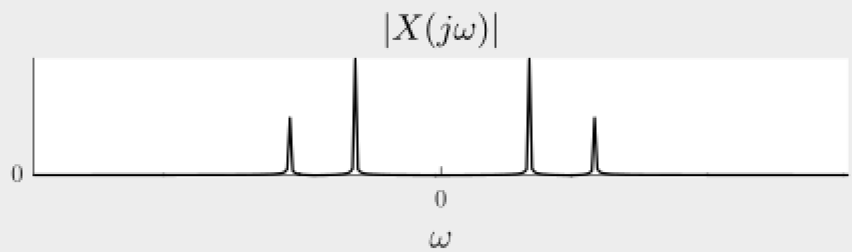
There are many different types of windows, which might be chosen according to desired purposes. [link] demonstrates how different windows can have varying results in terms of a particular application.

Example:

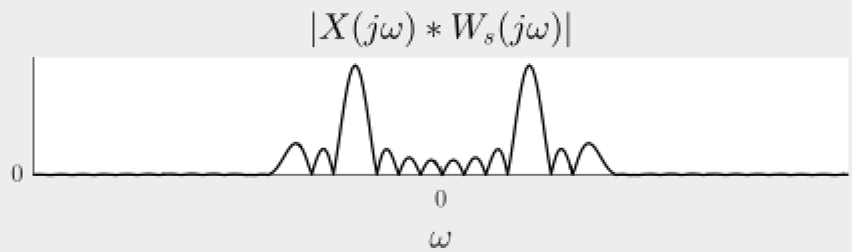
Windows for Sinusoidal Resolution

We will now see how the choice of windows can make a difference in certain applications. Suppose $x(t)$ is a signal that is composed of two sinusoids ([link]). As we would expect, its CTFT is simply four spikes: two for each sinusoid ([link]).

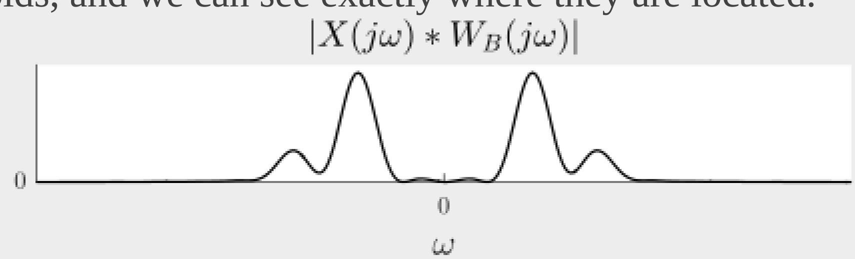




Suppose we use a rectangular window to extract a portion of a signal. [link] shows the CTFT of the result.



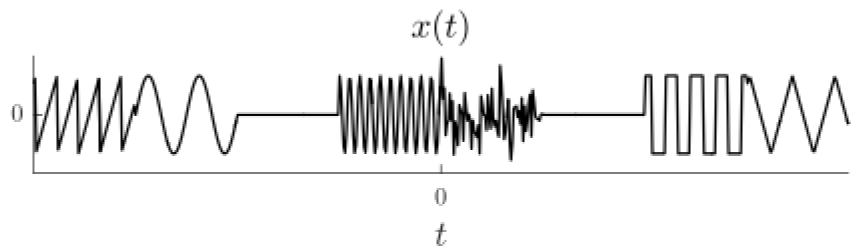
Based upon this CTFT and all of its ripples, it is very difficult to determine the frequency and number of sinusoids in $x(t)$! But now look at the result when a triangular window is used ([link]). Now it is clear there were only two sinusoids, and we can see exactly where they are located.



Windowing Application: Short-time Fourier Transform

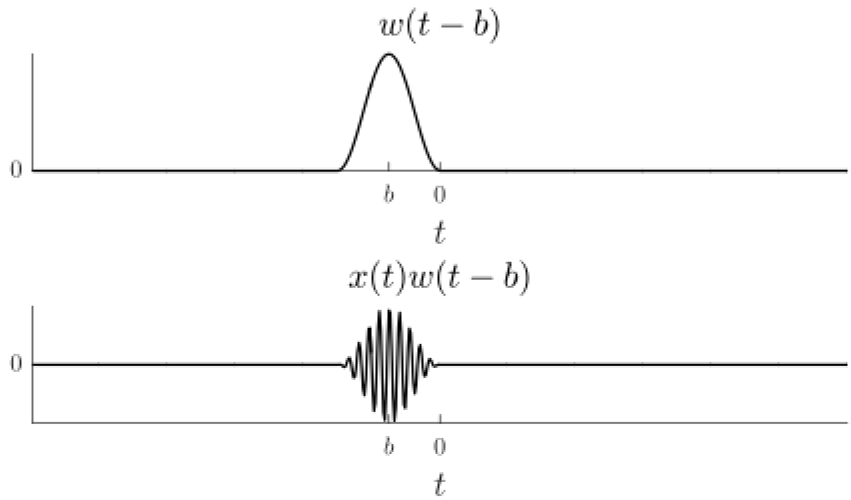
A CTFT looks at the frequency components of an entire infinite-length signal. There may be some instances in which we are more interested in the frequency content at a particular point in a signal, such as the one in [link]. For signals like this, in which the frequency content varies over the course of a signal, it would be helpful to have a way to represent such changes.

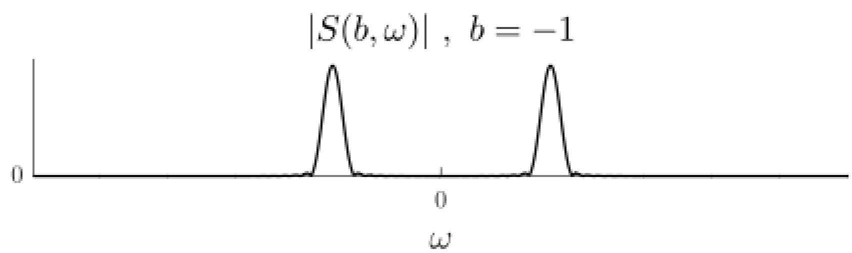
The frequency content at particular points in this signal varies considerably over time.



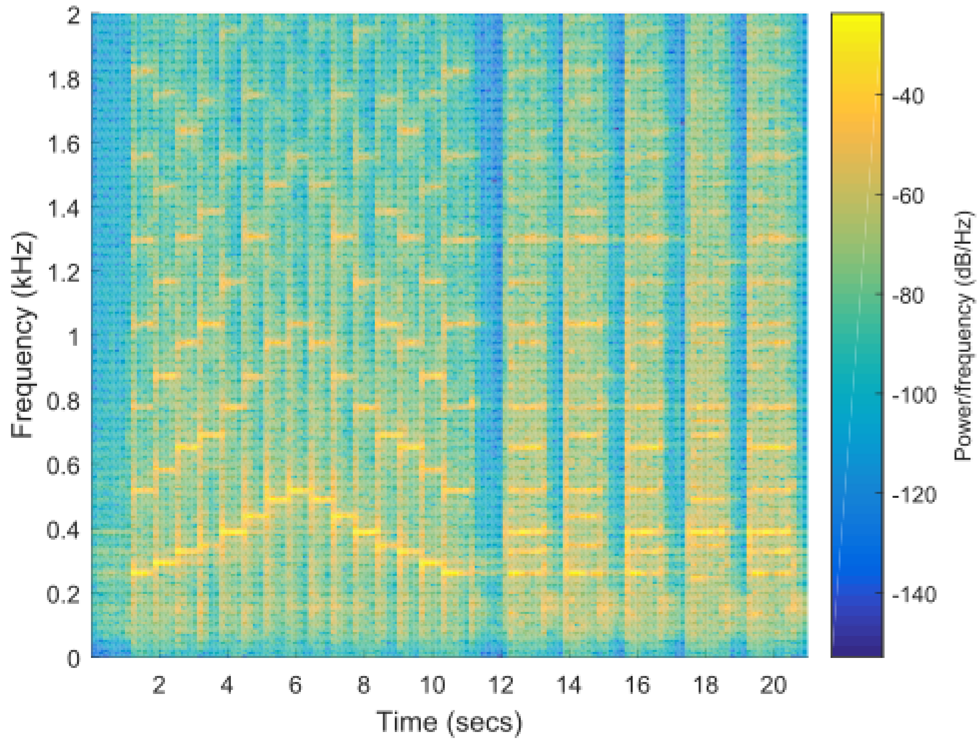
It turns out there is a way, and it is called the **short-time Fourier transform (STFT)**. The STFT works by windowing a signal at all of its time values and taking the CTFT of the windowed signal at each point.
$S(b,\omega)=\int_{-\infty}^{\infty}x(t)w(t-b)e^{-j\omega t}dt$
So we start with the original signal of [link], then we multiply it by a window, in this case centered at $b=-1$ ([link]). At this point, the CTFT of the windowed signal is taken, giving the result for that particular time point b ([link]). We can see from the CTFT that at that point in the signal, it is simply a sinusoid.

The moving window extracts a portion of the signal at all points in time (here, at $t=-1$).





Of course, the STFT is defined for many different time points. Representing it therefore requires using a third dimension in plots (b is one dimension, ω is another, and the value $S(b,\omega)$ is the third). When $|S(b,\omega)|^2$ is used, and in particular when its value is represented as a color, then the plot is called a **spectrogram**. [link] is a spectrogram of an audio recording of a scale and then a series of chords on a piano.



Each vertical column of the spectrogram represents the CTFT of a the signal windowed at a particular time b. We can see the frequencies increasing and then decreasing in time as the scale is played. Each note is composed of its fundamental frequency (e.g., .440 kHz for the note "A") and a number of harmonic frequencies as well (these are what give different instruments their particular sound characteristics, even when playing the same note). If you look closely enough, you can see that each chord is a combination of three fundamental frequencies.

The Laplace Transform

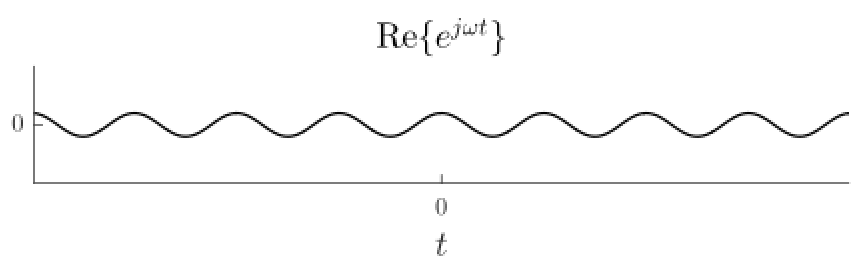
A Larger Class of Signals for Representation

In our study of continuous-time signals, we have seen that the complex sinusoid $e^{j\omega t}$ ([link]) is pretty special. One reason is that any signal $x(t)$ (as long as it satisfies the Dirichlet conditions) can be represented in terms of a weighted combination of complex sinusoids. The representation is weighted according to the CTFT:

$$X(j\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt$$

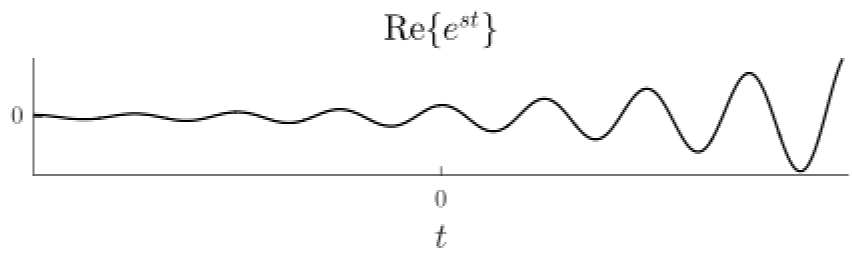
A second reason is that the complex sinusoid is an eigenfunction of LTI systems, meaning that if $e^{j\omega t}$ is input into an LTI system, the output will be $H(\omega)e^{j\omega t}$, where $H(\omega)$ is the CTFT of the system's impulse response.

The real part of a complex sinusoid.



It so happens that these two properties of complex sinusoids also apply to a larger class of signals, complex exponentials. A complex exponential is a signal e^{st}, where s is some complex number ([link]). Such signals are also eigenfunctions of LTI systems, and they can also be used to represent other signals.

The real part of a complex exponential.



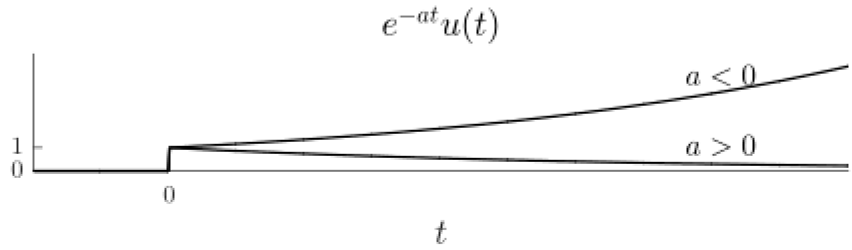
You may wonder why we would want to consider working with e^{st} when $e^{j\omega t}$ seemed to work perfectly well. The main reason is that it is a more helpful tool when it comes to continuous-time systems analysis and implementation. By being a larger class of signals, it can represent some signals better than the CTFT can. For example, there is no CTFT for the signal $e^t u(t)$, but we can represent it using complex exponentials. And some signals--such as the step function $u(t)$--have CTFTs that require the use of a delta function, while their complex exponential representations don't.

A Transform Using Complex Exponentials: The Laplace Transform

Just as complex sinusoids $e^{j\omega t}$ have the CTFT, there is also a frequency domain transform for complex exponentials e^{st}. It is called the **Laplace transform**. The Laplace transform for an infinite-length continuous time signal e^{st} is defined as:

$$H(s) = \int_{-\infty}^{\infty} h(t) e^{-st} dt$$

Technically, this is the "bilateral" Laplace transform because the integral is calculated from $t = -\infty$ to $t = \infty$ (there is also a "unilateral" one that goes only from $t = 0$ to $t = \infty$, but we won't be dealing with it). Let's find the Laplace transform of a signal $h(t) = e^{-at}u(t)$, where a is some real constant ([link]).



We will just follow the formula, with $s = \sigma + j\omega$:

$$\begin{aligned}
H(s) &= \int_{-\infty}^{\infty} h(t)e^{-st}dt \\
&= \int_{0}^{\infty} e^{-at}e^{-st}dt \\
&= \int_{0}^{\infty} e^{-(a+s)t}dt \\
&= \left[\frac{-1}{a+s}e^{-(a+s)t}\right]_0^{\infty} \\
&= \frac{-1}{a+s}\left[\lim_{t\to\infty} e^{-(a+s)t} - 1\right] \\
&= \frac{-1}{a+s}\left[\lim_{t\to\infty} e^{-(a+(\sigma+j\omega))t} - 1\right] \\
&= \frac{-1}{a+s}\left[\lim_{t\to\infty} e^{-j\omega t}e^{-(a+\sigma)t} - 1\right]
\end{aligned}$$

At this point, we have to acknowledge that the term $\lim_{t\to\infty} e^{-j\omega t}e^{-(a+\sigma)t}$ does not exist for all values of s. The magnitude of $e^{-j\omega t}$ is always 1, but as $t \to \infty$, $e^{-(a+\sigma)t}$ will either tend to 0 if $\sigma > -a$ or become unbounded if $\sigma < -a$. The limit will obviously exist in the first case, and will not for the latter. So then we have:

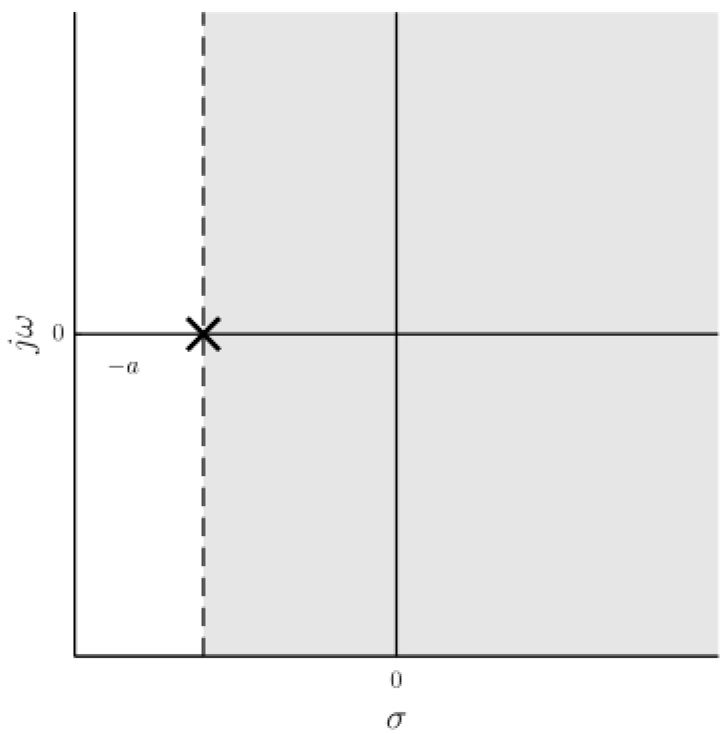
$$H(s) = \begin{cases} \frac{1}{s+a} & \mathrm{Re}\{s\} > -a \\ \text{undefined} & \mathrm{Re}\{s\} \leq -a \end{cases}$$

However, this is typically shortened to be $H(s) = \frac{1}{s+a}$, $\mathrm{Re}\{s\} > -a$, with the implied understanding it is undefined for $\mathrm{Re}\{s\} \leq -a$.

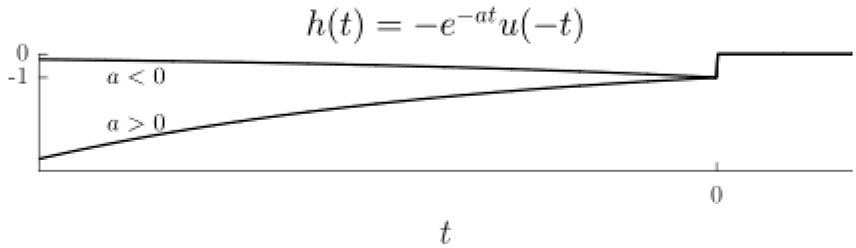
The Region of Convergence

The Laplace transform we calculated for the signal $h(t) = e^{-at}u(t)$ was found to be $\frac{1}{s+a}$, $\mathrm{Re}\{s\} > -a$. The values of s for which the Laplace transform exists is known as the **region of convergence (ROC)**. We can also plot the ROC as a shaded region. All the values of s (recalling the complex-valued s can be expressed $\sigma + j\omega$) in the ROC are shaded in [link].

The shaded parts of the s-plane are the values of s for which a given signal's Laplace transform exists; these values are known as the Laplace transform's Region of convergence (ROC).



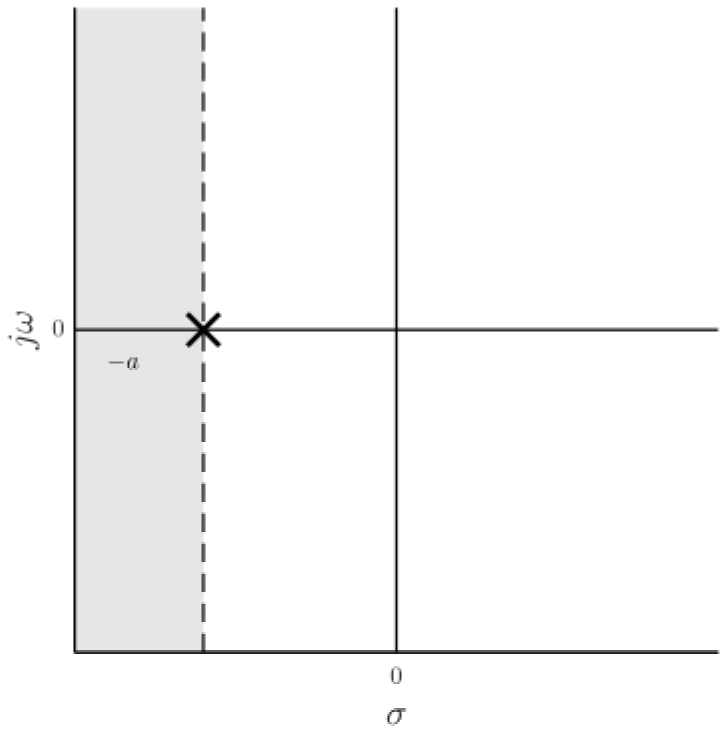
The "x" on the plot above is a "pole" of $H(s)$, but we'll consider that more fully when discussing the properties of the Laplace transform. Note that it is very important to include the s values of the ROC (if applicable) when expressing a Laplace transform, for two different signals might have what appears to be the same Laplace transform, albeit with a different ROC. Consider the signal $h(t) = -e^{-at}u(-t)$ ([link]).



Its Laplace transform is nearly identical to that of the first signal we considered, with one important difference:

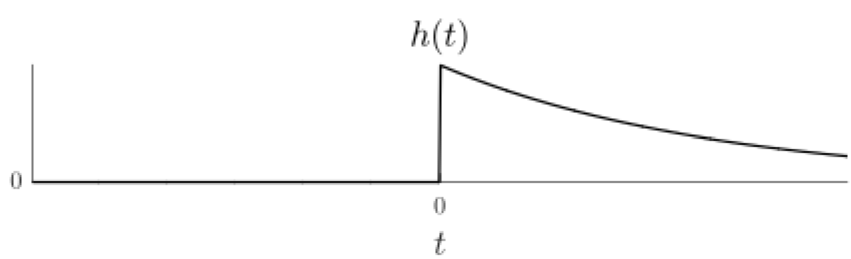
$$\begin{aligned} H(s) &= \int_{-\infty}^{\infty} h(t) e^{-st} dt \\ &= \int_{-\infty}^{0} e^{-at} e^{-st} dt \\ &= \int_{-\infty}^{0} e^{-(a+s)t} dt \\ &= \frac{-1}{s+a} \left[e^{-(a+s)t} \right]_{-\infty}^{0} \\ &= \frac{-1}{s+a} [1 - \lim_{t \to -\infty} e^{-(a+s)t}] \\ &= \frac{1}{s+a} \text{ , } \mathrm{Re}\{s\} < -a \end{aligned}$$

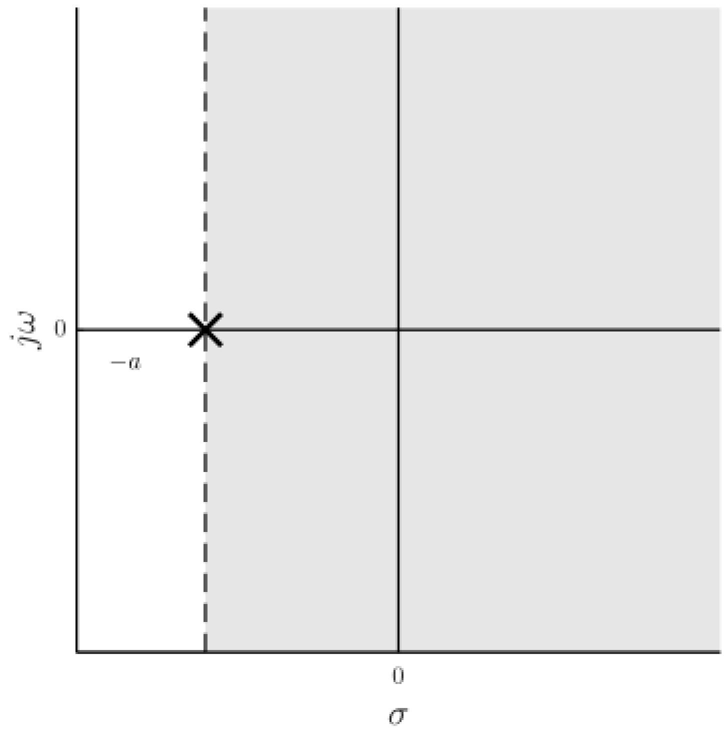
The only difference between this Laplace transform and the previous one is the ROC ([link]).



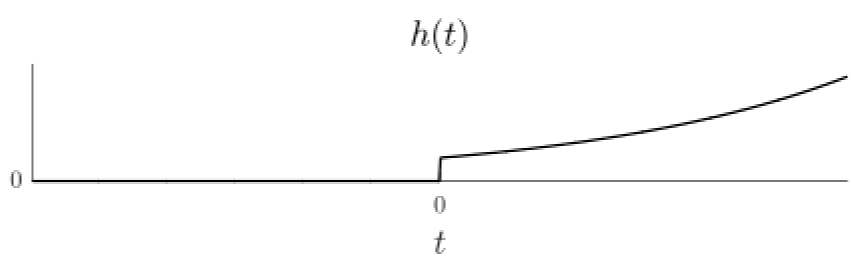
Note also how the ROC corresponds to the kind of signal. As we saw above, if a one-sided signal extends indefinitely to the right, its ROC does as well. The signal in [link] extends to $t \to \infty$ while decreasing, and the ROC of its Laplace transform also extends to the right indefinitely.

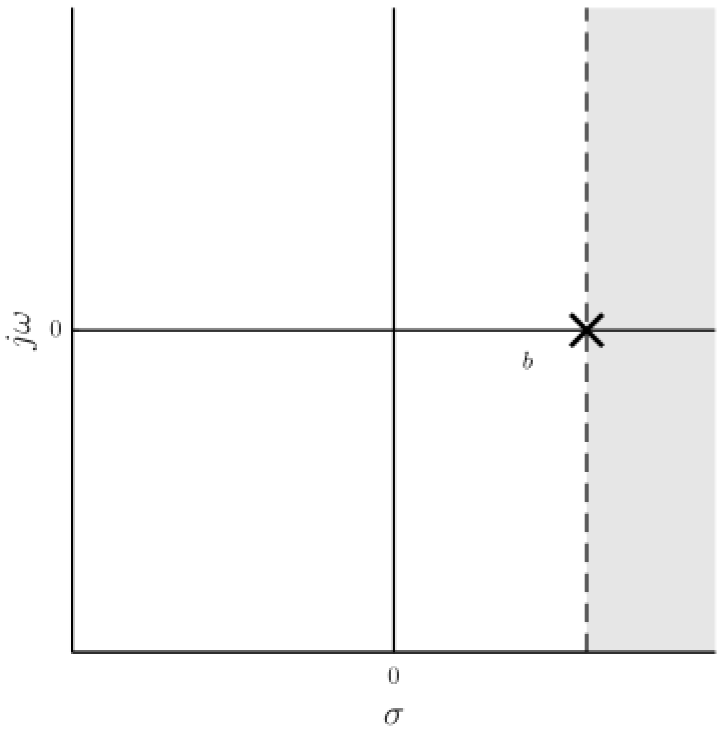
A one-sided signal that extends indefinitely to the right (i.e., $t \to \infty$) has a ROC that also extends indefinitely to the right.





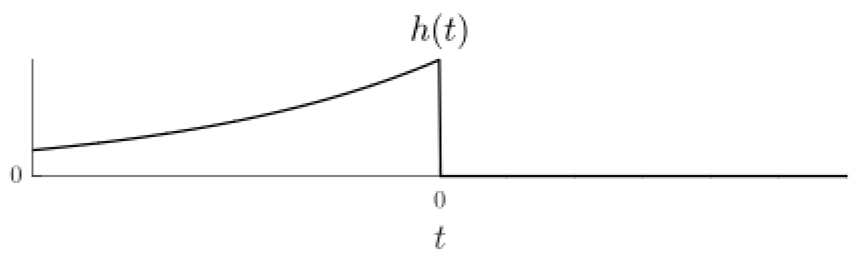
If it increases while extending $t \to \infty$, the ROC still also extends indefinitely to the right ([link]).

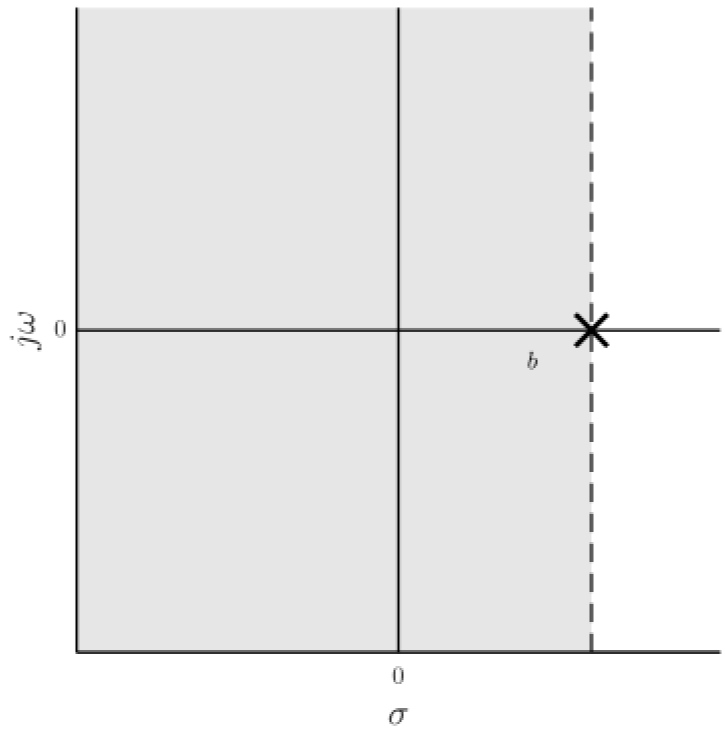




Likewise, when a one-sided signal extends to $t \to -\infty$, the ROC extends indefinitely to the left ([link]).

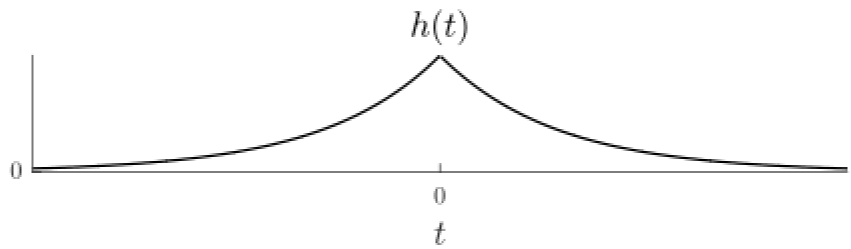
A one-sided signal that extends indefinitely to the left (i.e., $t \to -\infty$) has a ROC that also extends indefinitely to the left.

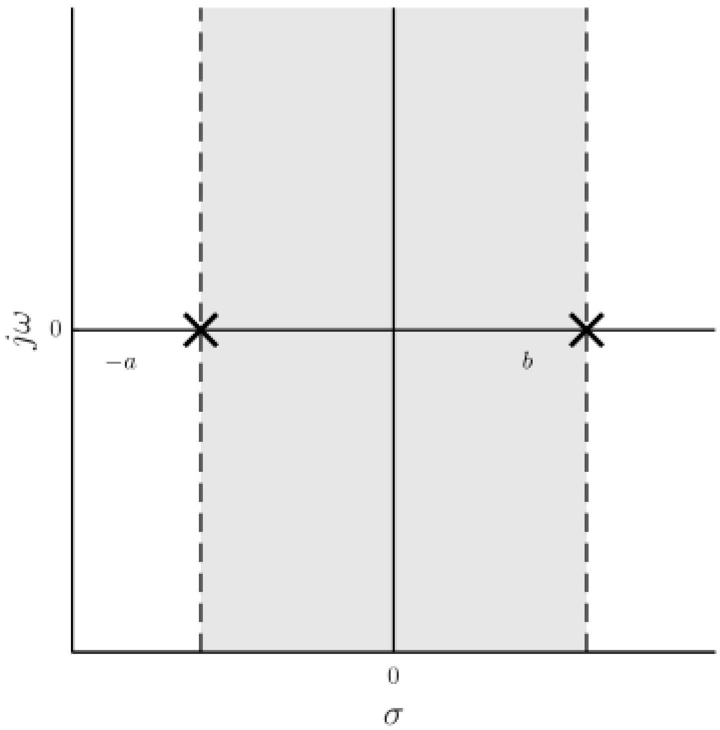




Two-sided signals (those that extend to both $t \to -\infty$ and $t \to \infty$) do not always converge, but when they do, the ROCs of their Laplace transforms extend between the poles ([link]).

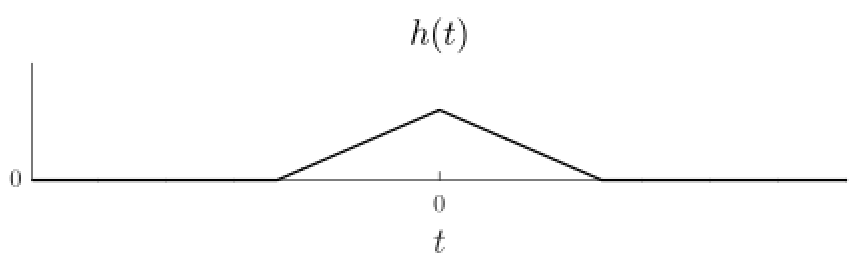
If a two-sided signal (nonzero from $t \to -\infty$ to $t \to \infty$) has a Laplace transform, it will not extend indefinitely, but rather will exist between two poles.

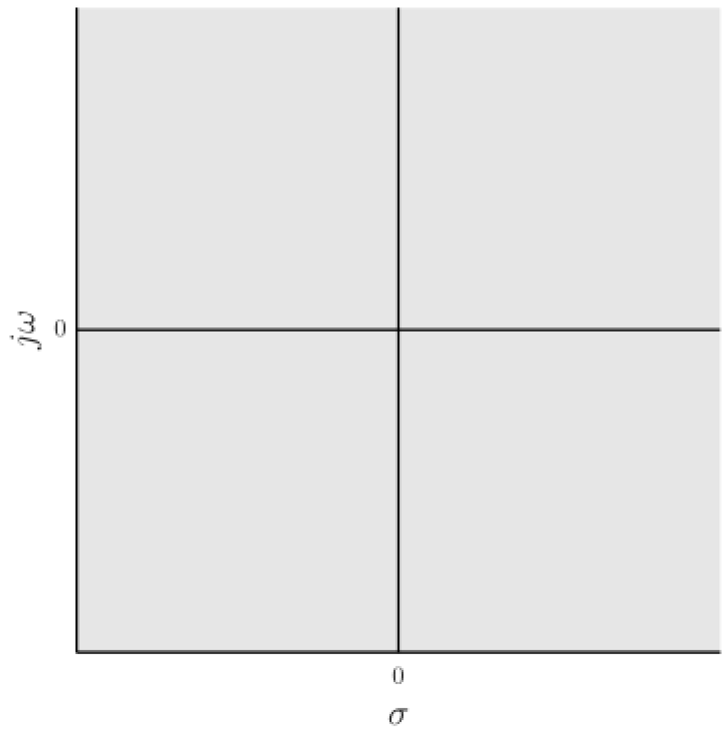




Finally, if a signal is nonzero for only a finite section of time, then its ROC will be the entire s-plane ([link]).

Signals that are nonzero (and bounded) for a finite amount of time will have Laplace transforms whose ROCs are the entire s-plane.





So if a Laplace transform has a ROC, it is necessary to know it in order to recover the original signal, a task to which we will now turn.

Inverting the Laplace Transform

Recall the Laplace transform definition for a signal $h(t)$:

$$H(s) = \int_{-\infty}^{\infty} h(t)e^{-st}dt$$

Given $H(s)$, it is natural to inquire as to how we may then recover $h(t)$. Unlike with the CTFT or CTFS, we cannot perform a simple integration -- instead we would need to calculate a contour integral. Rather than take that approach, it is usually more common to do something a bit more intuitive.

Suppose we are given $H(s) = \frac{1}{s+3}$, $\mathrm{Re}\{s\} > -3$. We have already seen above that a Laplace transform of $\frac{1}{s+a}$, $\mathrm{Re}\{s\} > -a$ corresponds to $e^{-at}u(t)$, so all we have to do is set $a = 3$, which gives us $h(t) = e^{-3t}u(t)$
.

Yes, that was a simple example. Suppose now that we have something a little more complicated, that we want to find the inverse Laplace transform for some $H(s) = H_1(s) + H_2(s)$, and we already know $h_1(t)$ and $h_2(t)$. Since integrals (and contour integrals) are linear operators, then the inverse will likewise be $h_1(t) + h_2(t)$. Of course, it will be helpful to first have a table of elementary Laplace transform pairs ([link]).

$h(t)$	$H(s)$	ROC
$\delta(t)$	1	all s
$\delta(t - t_0)$	e^{-st_0}	all s
$u(t)$	$\frac{1}{s}$	$\mathrm{Re}\{s\} > 0$
$-u(-t)$	$\frac{1}{s}$	$\mathrm{Re}\{s\} < 0$
$e^{-at}u(t)$	$\frac{1}{s+a}$	$\mathrm{Re}\{s\} > -\mathrm{Re}\{a\}$
$-e^{-at}u(-t)$	$\frac{1}{s+a}$	$\mathrm{Re}\{s\} < -\mathrm{Re}\{a\}$
$\frac{t^{n-1}}{(n-1)!}e^{-at}u(t)$	$\frac{1}{(s+a)^n}$	$\mathrm{Re}\{s\} > -\mathrm{Re}\{a\}$
$\frac{-t^{n-1}}{(n-1)!}e^{-at}u(-t)$	$\frac{1}{(s+a)^n}$	$\mathrm{Re}\{s\} < -\mathrm{Re}\{a\}$

Given this table, the task of inverting the Laplace transform is merely one of breaking up the $H(s)$ expression into a sum of parts that resemble elements on the table. Because $H(s)$ is often expressed as a fraction with a polynomial numerator and denominator, this will typically involve finding the roots of the denominator and then performing a "partial fractions"

expansion. The easiest way to explain partial fractions is to just go through an example ([link]).

Example:

Suppose we are given an $H(s)$ of the form $\frac{2s^2+9s+5}{s^2+4s+3}$ with a ROC of $\text{Re}\{s\} > -1$. We would like to simplify it so that it looks more like, e.g., $\frac{A}{s-a} + \frac{B}{s+b}$.

The first step is to ensure that the polynomial power of the numerator is less than that of the denominator. In our example it is not, so we need to do a polynomial long division. It is just like long division with numbers, except it goes by term instead of by digit. See below:

$$\begin{array}{r|l} & \quad\quad\quad\;\; 2 \\ s^2+4s+3\;) & 2s^2+9s+5 \\ & 2s^2+8s+6 \\ & \quad\quad\;\; s-1 \end{array}$$

So we have that $H(s) = 2 + \frac{s-1}{s^2+4s+3}$. Now we will look to simplify the remainder term. The second step of partial fractions is to find the roots of the denominator. In this case, $s^2+4s+3 = (s+1)(s+3)$. The third step is to then express this as a sum of two simpler terms:

$\frac{s-1}{(s+1)(s+3)} = \frac{A}{s+1} + \frac{B}{s+3}$

Now we just need to solve for A and B:

$$\frac{s-1}{(s+1)(s+3)} = \frac{A}{s+1} + \frac{B}{s+3}$$
$$s-1 = \frac{A(s+1)(s+3)}{s+1} + \frac{B(s+1)(s+3)}{s+3}$$
$$s-1 = A(s+3) + B(s+1)$$
$$s-1 = (A+B)s + (3A+B)$$
$A + B = 1\ ,\ 3A + B = -1 \rightarrow A = -1\ ,\ B = 2$
$$\frac{s-1}{(s+1)(s+3)} = \frac{-1}{s+1} + \frac{2}{s+3}$$
This then gives us $H(s) = 2 + \frac{-1}{s+1} + \frac{2}{s+3}$. Recalling that the ROC is $\mathrm{Re}\{s\} > -1$ and using the table above, we have:
$h(t) = 2\delta(t) - e^{-t}u(t) + 2e^{-3t}u(t)$

The only wrinkle in this process is if a root is repeated, in which case it needs to be decomposed through partial fractions like this:
$\frac{1}{(s-a)^N} = \frac{A_N}{(s-a)^N} + \frac{A_{N-1}}{(s-a)^{N-1}} + \cdots + \frac{A_1}{s-a}$

Laplace Transform Properties

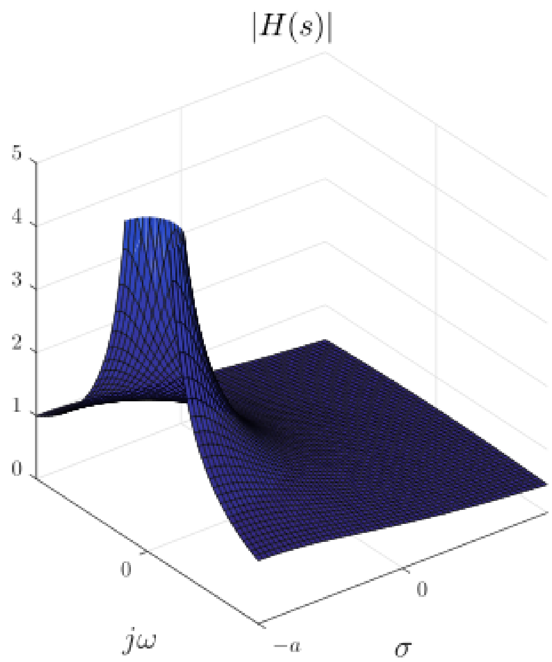
Visualizing the Laplace Transform

Recall the definition of the Laplace transform $H(s)$ for an infinite-length continuous time signal $h(t)$:

$$H(s) = \int_{-\infty}^{\infty} h(t)e^{-st}dt$$

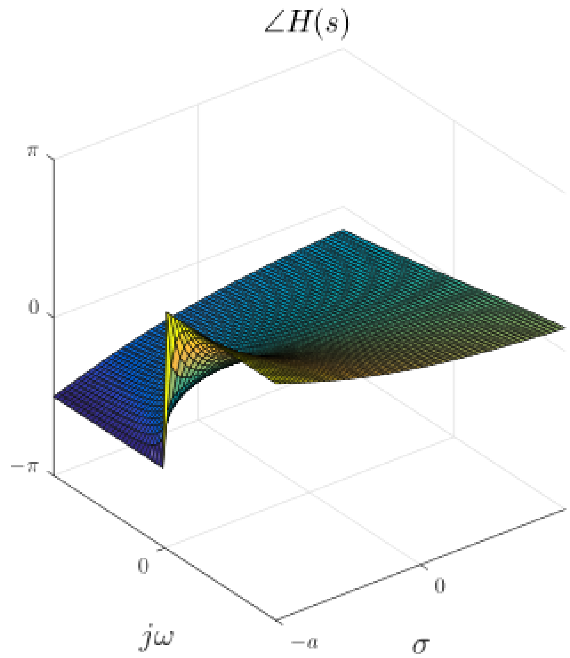
As we consider the properties of the Laplace transform, one of the first things to note is that $H(s)$ is a complex-valued function of a complex-valued variable. Now the CTFT of $h(t)$, $H(j\omega)$, is a complex-valued function of a real variable. As such, it is straightforward to plot, say, the magnitude $|H(j\omega)|$; it is a simple two-dimensional plot. It is trickier to plot $H(s)$, because even if we limit ourselves to plotting just its magnitude, $|H(s)|$, the dependent variable s is two dimensional (having both a real and imaginary part), so three dimensions are required. Suppose $H(s) = \frac{1}{s+a}$, $\mathrm{Re}\{s\} > -a$. Then $|H(s)|$ would be plotted as shown in [link].

Even if we only plot the magnitude of a Laplace transform, three dimensions are required to plot it.



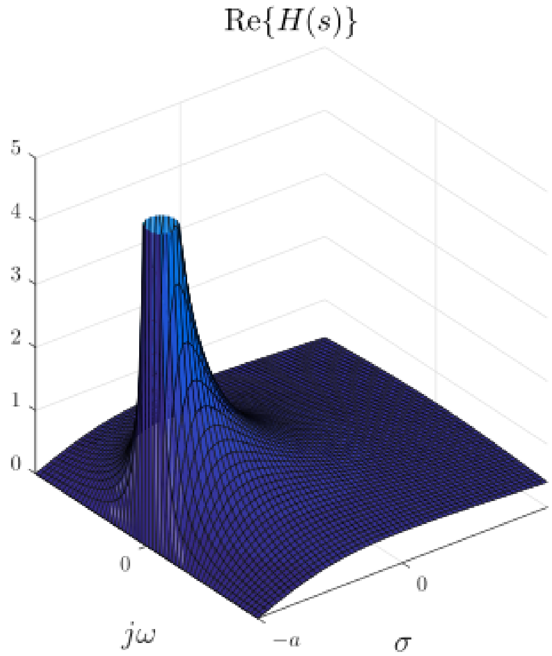
To complete our representation of $H(s)$, [link] is a plot of $\angle H(s)$.

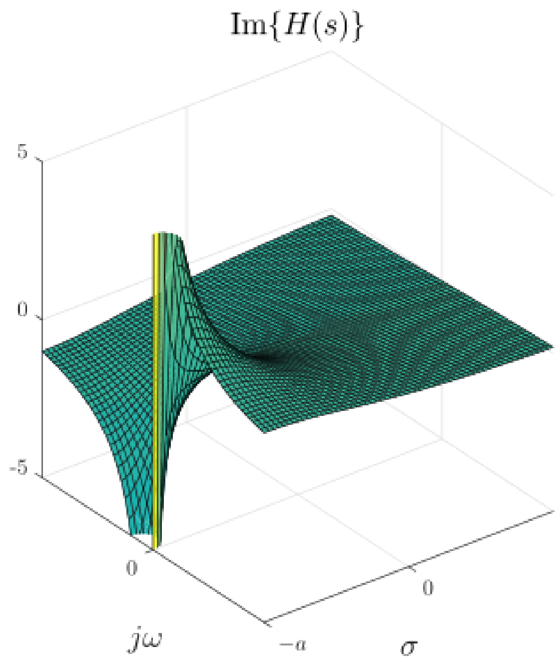
Plotting only the angle of a Laplace transform also requires three dimensions.



Of course, as with the CTFT, the real and imaginary plots of $H(s)$ could also have been plotted instead ([link]).

Instead of plotting magnitude and phase, a Laplace transform can be plotted in terms of its real and imaginary parts. As with the magnitude and phase, three dimensional plots are necessary due to s having both a real and an imaginary component.



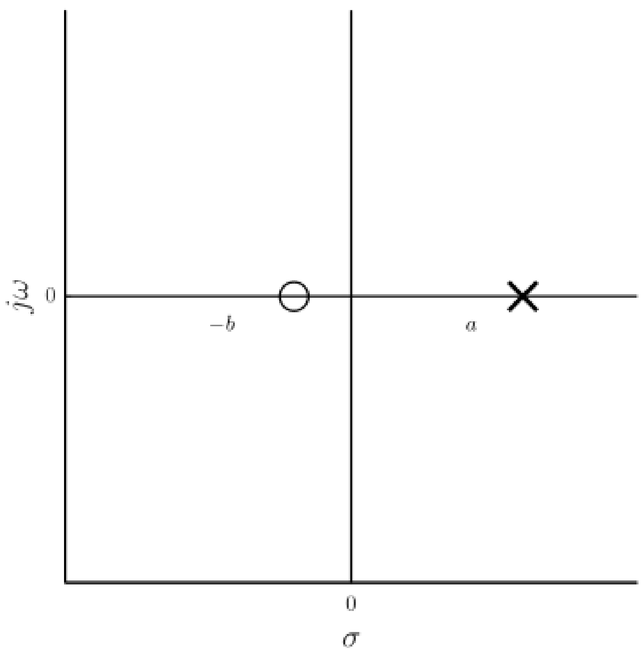


$H(s)$ is rarely plotted in this way. Rather than plot all of either the magnitude or phase or real part or imaginary part of $H(s)$ on the s plane (thus needing three dimensions), typically only the locations for which $H(s)$ and $\frac{1}{H(s)}$ are zero are plotted. These locations are known, respectively, as the zeros and poles of $H(s)$.

Poles and Zeros

Let us express $H(s)$ as a fraction, e.g., $H(s) = \frac{N(s)}{D(s)}$. Values of s for which $N(s)$ is zero are known as **zeros** of $H(s)$, and values of s for which $D(s)$ is zero are known as **poles** of $H(s)$. It's easy to remember which is which. Zeros are the points where $H(s) = 0$, and as you can see in [link], poles are the locations where $H(s)$ looks like, well, a pole (the real reason for the name, though, has to do with where ∞ is located on something called a "Riemann sphere").

So as nice as three dimensional plots of $H(s)$ are to look at, what matters most for a Laplace transform are its zeros and poles. It only takes a two dimensional plot to display these, using an "o" for a zero location and "x" for a pole location. So for example, if $H(s) = \frac{s+b}{s-a}$, the pole/zero plot would be as shown in ([link]).



The Transfer Function of LTI Systems

The Laplace transform is an interesting way to represent any continuous-time signal, but the most important signal it represents is that of the impulse response $h(t)$ of LTI systems (hence our frequent use of $H(s)$ in the discussion of Laplace transforms so far). The Laplace transform $H(s)$ of an LTI system's impulse response is so important, it has its own name: it is called the **transfer function** of the system.

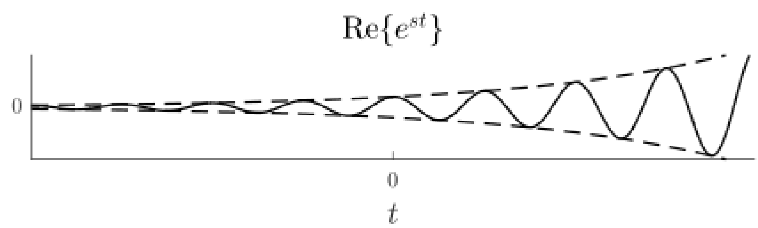
The transfer function is a straightforward way of describing how an LTI system works. Supposing e^{st} is input into a system with a transfer function $H(s)$, the output

will be $H(s)e^{st}$. $H(s)$ tells us how the system modifies various complex exponential inputs (and remember that most inputs can be represented as an infinite combination of such complex exponentials). Splitting s into $\sigma + j\omega$ and simplifying a little, we can see more clearly what it does:

$$\begin{aligned} H(s)e^{st} &= |H(s)|e^{\angle H(s)}e^{st} \\ &= |H(s)|e^{\angle H(s)}e^{(\sigma+j\omega)t} \\ &= |H(s)e^{\sigma t}|e^{j(\omega t+\angle H(s))} \end{aligned}$$

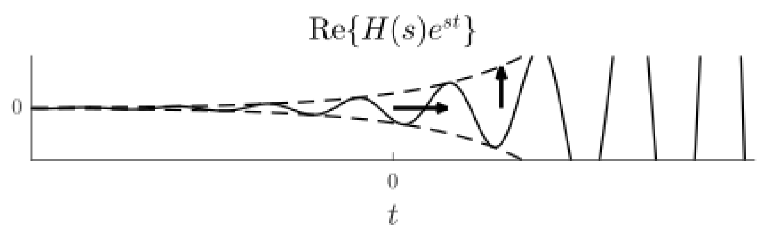
Recall that a complex exponential e^{st} is simply a complex sinusoid with an exponential envelope ([link]).

A complex exponential e^{st} is a complex sinusoid with an exponential envelope.



When this complex exponential is input into an LTI system, $|H(s)|$ scales the envelope, and $\angle H(s)$ introduces a phase shift on the sinusoid ([link]).

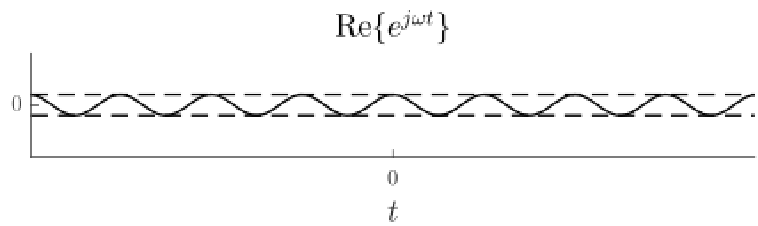
The arrows in this figure illustrate how the transfer function of an LTI system modifies the phase and magnitude of complex exponential inputs.



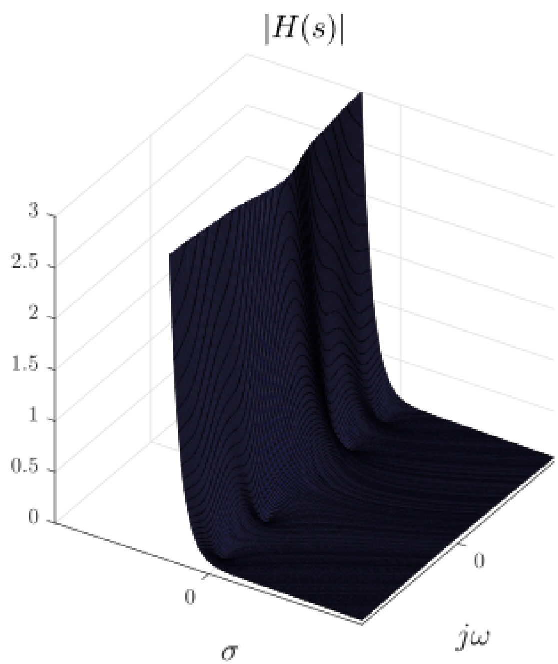
So for LTI systems, $H(s)$ takes complex exponential inputs and scales them. You may remember from when we covered the CTFT that this operation is similar to what $H(j\omega)$ does to complex sinusoids. That is because $H(j\omega)$ is just a special case of $H(s)$. If, for $s = \sigma + j\omega$, we let $\sigma = 0$, then we have for an input $e^{j\omega t}$, the output is: $H(j\omega)e^{j\omega t}$

$H(j\omega)$ is called an LTI system's **frequency response**, because it tells us how the system modifies (or "responds") to inputs of different frequencies. If $H(s)$ tells us how the system responds to various e^{st}, $H(j\omega)$ tells us how it responds so just a certain subset of those, namely, those without an exponential envelope ([link]).

If a complex exponential has a constant-valued envelope, then it is also a complex sinusoid.



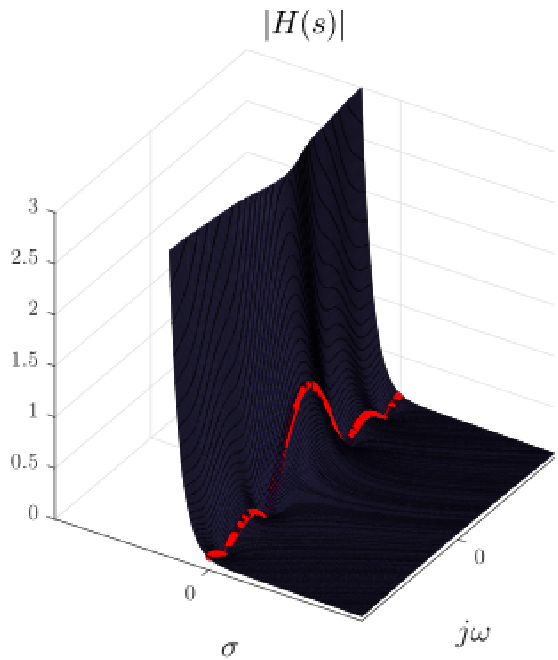
The magnitude of scaling that $H(s)$ imparts on different complex exponentials can be seen on a plot of it. Consider, for example, a system with transfer function $H(s) = \frac{1-e^{-s}}{s}$ ([link]).



As indicated earlier, a three dimensional plot can be difficult to produce or visualize. The frequency response considers only the part of the plot for which $\sigma = 0$ ([link]).

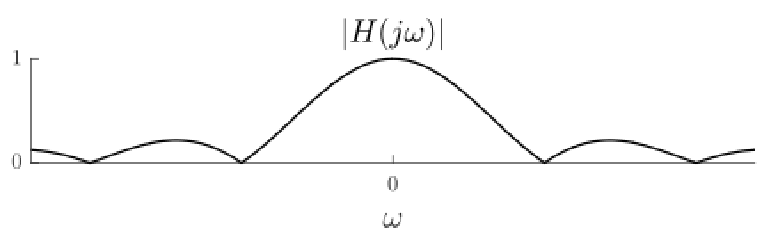
Corresponding inputs are complex sinusoids.

For a given transfer function we can focus on the part where $\sigma = 0$, which is highlighted here as a red line.



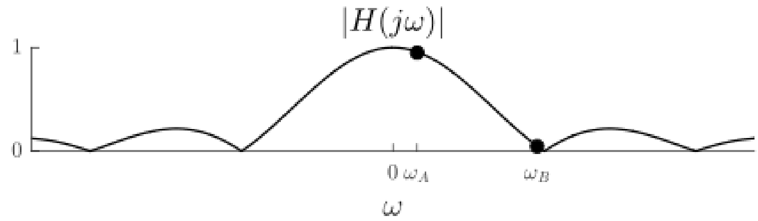
That line is only two-dimensional, and therefore there is a simpler way to plot it ([link]).

We can isolate the $\sigma = 0$ portion of a transfer function and then plot it in two dimensions.



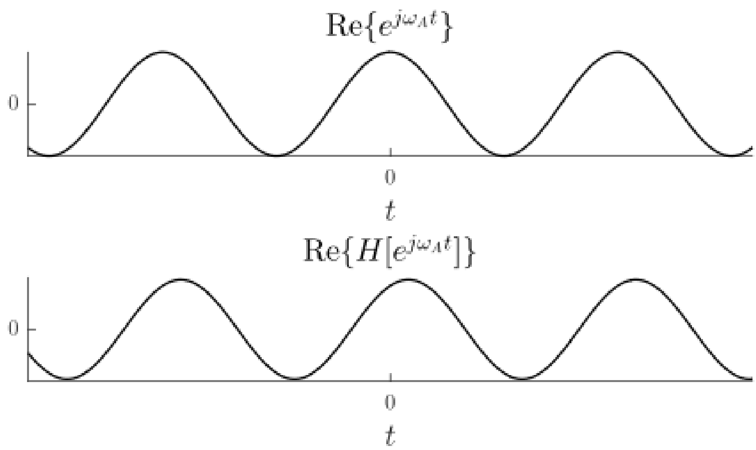
It is easier to see how it modifies different input frequencies, say a low frequency like ω_A or a higher frequency like ω_B ([link]).

The magnitude of the frequency response shows how an LTI system scales inputs of various frequencies.



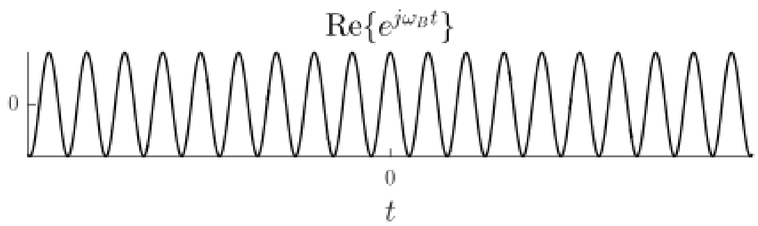
Low frequency inputs, such as ω_A, are mostly unchanged. The output for this signal is only very slightly attenuated, with a phase shift ([link]).

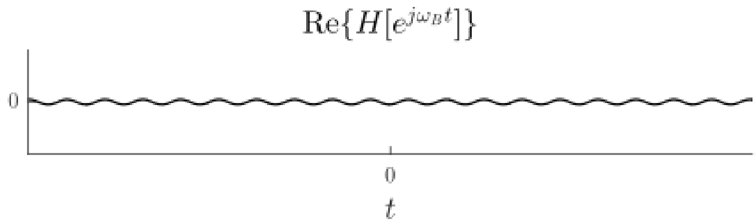
A low frequency input is only slightly attenuated by the system.



But things change with the higher frequency input ω_B. The system strongly attenuates it, giving the output shown in [link].

A high frequency input is strongly attenuated by the system.



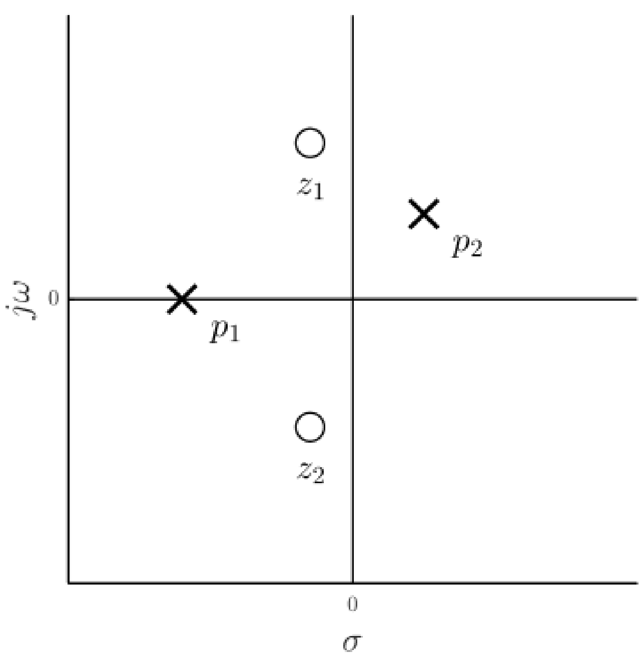


Given how it is more convenient to visualize the input/output relationship with the frequency response $H(j\omega)$, you may wonder what the point of using the transfer function $H(s)$ may be. As mentioned before, $H(s)$ can mathematically represent some functions more easily than the CTFT. Additionally, it is somewhat more concise to deal with the variable s than $j\omega$. And finally, as we will see when we discussion implementation of CT systems with the Laplace transform, it is straightforward to represent analog circuit elements with the variable s.

Poles/Zeros and the Transfer Function

Zeros are locations in the s domain where the numerator of $H(s)$ is zero, and poles are the locations where the denominator is zero. These locations are often plotted on the two-dimensional s-plane, e.g. as shown in [link].

The poles and zeros of a Laplace transform are often plotted in the two-dimensional s-plane



Suppose $H(s)$ is factored out, so that we can easily see what all the poles and zeros are:

$$H(s)=\frac{(s-z_1)(s-z_2)\cdots(s-z_N)}{(s-p_1)(s-p_2)\cdots(s-p_N)}$$

If we would like to know what the magnitude of $H(s)$ is for a given value of s, we have that:

$$|H(s)|=\frac{|s-z_1||s-z_2|\cdots|s-z_N|}{|s-p_1||s-p_2|\cdots|s-p_N|}$$

Let's concentrate on each individual term, e.g. $|s-z_1|$. Defining $s=\sigma+j\omega$ and $z_1=a+jb$, then we have that

$$s-z_1=(\sigma-a)+j(\omega-b)$$

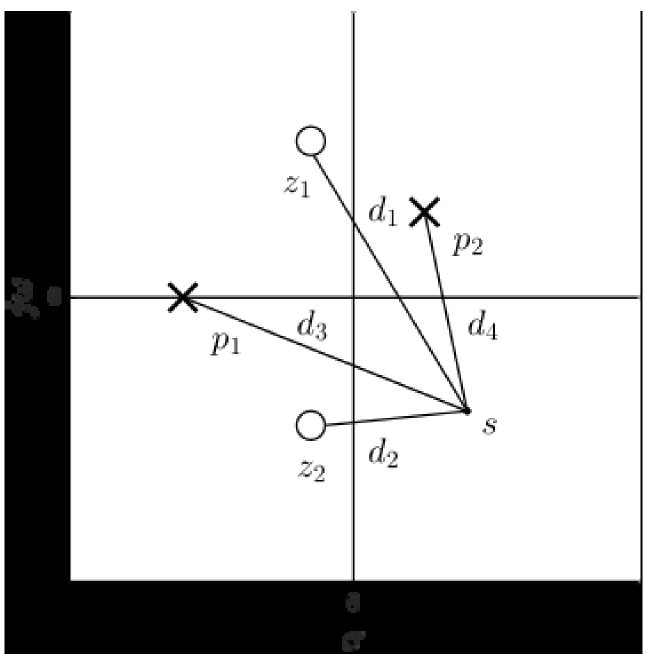
and therefore

$$|s-z_1|=\sqrt{(\sigma-a)^2+(\omega-b)^2}$$

And this is simply the euclidean distance from s to z_1.

So then, the magnitude of the transfer function $H(s)$ at point s is the product of the distances to zeros divided by the product of the distances to poles. That is to say, the closer s is to zeros, the smaller $|H(s)|$ will be, and the closer it is to poles, the larger it will be. Consider the pole-zero plot and s location in [link].

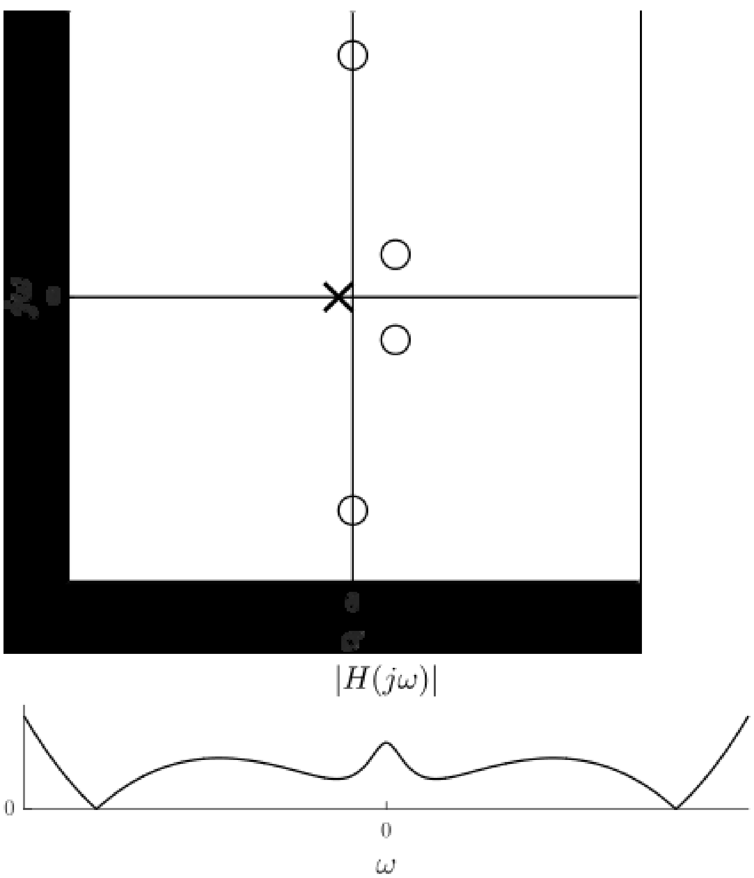
The magnitude of the transfer function $H(s)$ at point s is the product of the distances to zeros divided by the product of the distances to poles.



The magnitude of $H(s)$ at that point s is $|H(s)| = \frac{d_1 d_2}{d_3 d_4}$. To get a general idea of all of $|H(s)|$, we mentally move s around in the plane and see where it is in relation to the various poles and zeros.

As we have already seen, though, in practice we typically are more concerned with visualizing the magnitude of the frequency response, $|H(j\omega)|$. Since $H(j\omega)$ is simply $H(s)$ with $s = j\omega$, we can sketch it by moving s up and down the $j\omega$ axis, seeing how close it is to poles and zeros at different points. Take, for instance, the pole-zero plot in [link].

The magnitude of a system's frequency response can be visualized by noting the distance away from zeros and poles for every point on the $j\omega$ axis.



Moving just around the $j\omega$ axis, s is very close to a pole at $\omega = 0$, so we expect the frequency response to be relatively large there. Then as we move closer to zeros, especially where they touch the axis, the frequency response will decrease to 0. But

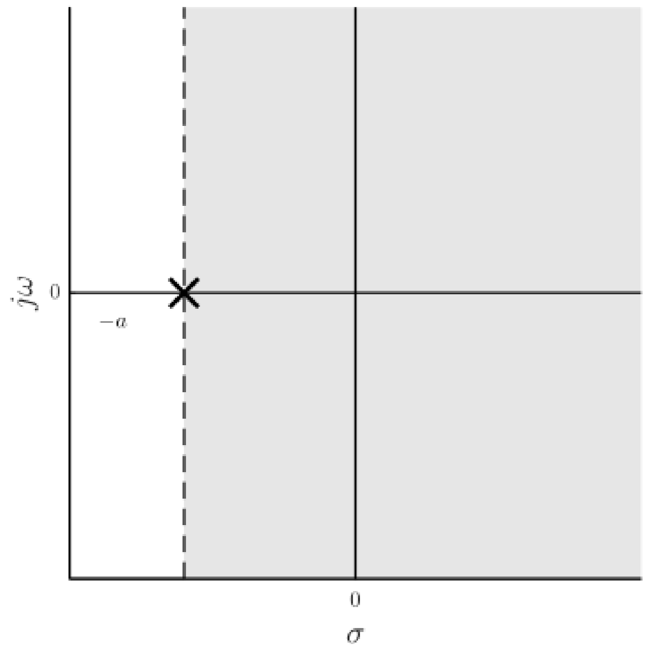
then as ω continues to increase, it gets farther and farther from more zeros (four) than poles (only one), so the frequency response will increase again.

Stability, Causality, and the Transfer Function

We have seen previously that we can determine if an LTI system is causal or stable from looking at its impulse response. If the impulse response $h(t)$ is 0 for all $t < 0$, then the system is causal. If $\int_{-\infty}^{\infty} |h(t)| dt < \infty$, then the system is BIBO stable. Since the transfer function $H(s)$ is simply the Laplace transform of $h(t)$, then it follows we should also be able to determine causality and stability simply from looking at the transfer function.

ROC and Causality
And indeed we can. We'll start with causality: an LTI system is causal if and only if its transfer function has a ROC that exists as $\text{Re}\{s\} \to \infty$. Or, put another way, the transfer function of a causal system extends indefinitely to the right. For example, consider a system whose impulse response is $h(t) = e^{-at}u(t)$. It is obviously causal, since the $u(t)$ ensures it is 0 for all $t < 0$. Its transfer function is $H(s) = \frac{1}{s+a}$, $\text{Re}\{s\} > \text{Re}\{-a\}$. This ROC clearly extends to the right, so $H(s)$ will exist as $\text{Re}\{s\} \to \infty$ ([link]).

The ROC of the transfer function of a causal LTI system will extend indefinitely to the right.



The proof of this property has two parts. First we need to show that for a causal system, $\lim\limits_{\text{Re}\{s\}\to\infty} H(s)$ exists. To do that, we use the fact that $h(t) = 0$ for $t < 0$ and that $H(s)$ exists at some point s_0. We'll also use an alternative formulation for the ROC being the region for which not only $H(s)$ exists, but $\int\limits_{-\infty}^{\infty} |h(t)e^{-st}|dt$ exists. This is a stronger requirement, but for all of the signals we'll consider the requirements are equivalent. So, we'll say that s_0 is in the ROC, and thus $\int\limits_{-\infty}^{\infty} |h(t)e^{-s_0 t}|dt$ is some number we'll call A. Now let's look at the magnitude of the transfer function as $\text{Re}\{s\} \to \infty$:

$$\begin{aligned}
\lim_{\mathrm{Re}\{s\}\to\infty} |H(s)| &= \lim_{\mathrm{Re}\{s\}\to\infty} \left| \int_{-\infty}^{\infty} h(t)e^{-st}dt \right| \\
&\le \lim_{\mathrm{Re}\{s\}\to\infty} \int_{0}^{\infty} |h(t)e^{-st}|dt \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} \int_{0}^{\infty} |h(t)e^{-s_0t}e^{-(s+s_0)t}|dt \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} \int_{0}^{\infty} |h(t)e^{-s_0t}||e^{-(s+s_0)t}|dt \\
&\le \int_{0}^{\infty} |h(t)e^{-s_0t}|1dt \\
&= A
\end{aligned}$$

Above we use the fact that $|e^{-(s+s_0)t}|$ will be less than or equal to 1 for all $\mathrm{Re}\{s\} > \mathrm{Re}\{s_0\}$. So $H(s)$ will exist as $\mathrm{Re}\{s\} \to \infty$.

Now we need to show that not only that $H(s)$ will exist as $\mathrm{Re}\{s\} \to \infty$ for causal systems, but for causal systems alone. To show that, we will show that the limit does not exist for acausal systems. All we need to do is recognize that $h(t)$ will be nonzero for some portion of the number line for $t < 0$, say $-a$ to $-b$ (where a and b are positive numbers, with $a > b$) and we'll call the minimum value of $|h(t)|$ for that portion B. Then we'll consider the convergence again:

$$
\begin{aligned}
\lim_{\mathrm{Re}\{s\}\to\infty} \int_{-\infty}^{\infty} |h(t)e^{-st}|dt &\geq \lim_{\mathrm{Re}\{s\}\to\infty} \int_{-a}^{-b} |h(t)e^{-st}|dt \\
&\geq \lim_{\mathrm{Re}\{s\}\to\infty} \int_{-a}^{-b} Be^{-st}dt \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} B\left[\frac{1}{-s}e^{-st}\right]_{-a}^{-b} \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} B\left[\frac{1}{-s}e^{bs} - \frac{1}{-s}e^{as}\right] \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} B\left[\frac{e^{bs}-e^{as}}{-s}\right] \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} B\left[\frac{e^{as}(e^{(b-a)s}-1)}{-s}\right] \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} B\left[\frac{e^{as}(0-1)}{-s}\right] \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} B\left[\frac{e^{as}}{s}\right] \\
&= \lim_{\mathrm{Re}\{s\}\to\infty} Bae^{as} \to \infty
\end{aligned}
$$

So $\mathrm{Re}\{s\} \to \infty$ is not in the ROC for acausal signals. Thus for an LTI to be causal, it is both necessary and sufficient that $\mathrm{Re}\{s\} \to \infty$ be in the ROC of its transfer function.

Because LTI systems are, in practice, almost always causal, the ROCs of their transfer functions are usually left unstated. The reason why is because, as we have just seen, the ROC of the transfer function of a causal system always extends indefinitely to the right. As a result, the ROC for a causal system extends to the right of the rightmost pole for the system.

ROC and BIBO Stability

We also recall that LTI systems with BIBO stability (a bounded input will always produce a bounded output) have impulse responses with finite L_1 norm ($\int_{-\infty}^{\infty} |h(t)|dt < \infty$).

The transfer function is also related to BIBO stability: an LTI system is BIBO stable if and only if $\mathrm{Re}\{s\} = 0$ is in the transfer function's ROC. We'll start by showing that if a system is BIBO stable, $\mathrm{Re}\{s\} = 0$ is in the transfer function's ROC; we'll do this by evaluating the Laplace transform of $h(t)$ at $\mathrm{Re}\{s\} = 0$:

$$\begin{aligned}
\left[\int_{-\infty}^{\infty} |h(t)e^{st}|dt\right]_{\mathrm{Re}\{s\}=0} &= \left[\int_{-\infty}^{\infty} |h(t)e^{(\sigma+j\omega)t}|dt\right]_{\sigma=0} \\
&= \int_{-\infty}^{\infty} |h(t)e^{(0+j\omega)t}|dt \\
&= \int_{-\infty}^{\infty} |h(t)e^{j\omega t}|dt \\
&\leq \int_{-\infty}^{\infty} |h(t)||e^{j\omega t}|dt \\
&= \int_{-\infty}^{\infty} |h(t)|dt \\
&< \infty
\end{aligned}$$

So, the Laplace transform of $h(t)$ exists at $\mathrm{Re}\{s\} = 0$, which by definition means that $\mathrm{Re}\{s\} = 0$ is in the ROC, and this is the case because the system is BIBO stable (and therefore $\int\limits_{-\infty}^{\infty} |h(t)|dt < \infty$). Thus for BIBO systems, $\mathrm{Re}\{s\} = 0$ is in the transfer function's ROC.

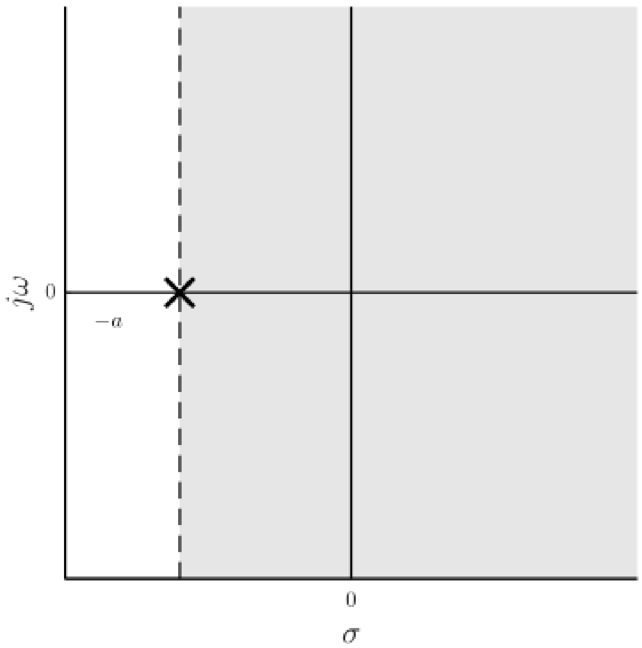
The converse is shown in a similar way. Suppose first that $\mathrm{Re}\{s\} = 0$ is in the transfer function's ROC, i.e. $\int\limits_{-\infty}^{\infty} |h(t)e^{st}|dt$ exists for $\mathrm{Re}\{s\} = 0$. It is straightforward to show that the L_1 norm of $h(t)$ also exists. A single step shows us why:

$$\int\limits_{-\infty}^{\infty} |h(t)e^{0+j\omega}|dt = \int\limits_{-\infty}^{\infty} |h(t)|dt$$

So if one exists, so must the other.

Putting all of this together, an LTI system is both causal and stable if and only if $\mathrm{Re}\{s\} \to \infty$ and $\mathrm{Re}\{s\} = 0$ are both in the transfer function's ROC. Or, graphically, it is causal and stable if its ROC extends from somewhere to the left of the $j\omega$ axis indefinitely to the right ([link]).

If a system's ROC contains the $j\omega$ axis and extends indefinitely to the right, then that system is both causal and stable.



Time/Frequency Relationship

The Laplace transform time/frequency properties are essentially the same as those for the CTFT, and are derived in the same way. The only difference is that the ROCs may also change when the signals/transforms change ([link]).

Property	$x(t)$	$X(s)$	ROC
Linearity	$\alpha x(t) + \beta g(t)$	$\alpha X(s) + \beta G(s)$	At least $R_X \cap R_G$
Time scaling	$x(at)$	$\frac{1}{\lvert a\rvert} X(\frac{s}{a})$	R scaled by a
Time shifting	$x(t - t_0)$	$e^{-st_0} X(s)$	R
Modulation	$e^{s_0 t} x(t)$	$X(s - s_0)$	R shifted by s_0

Time convolution	$x(t) * h(t)$	$X(s)H(s)$	At least $R_X \cap R_G$
Time differentiation	$\frac{d^n}{dt^n}x(t)$	$s^n X(s)$	At least R
s domain differentiation	$-tx(t)$	$\frac{d}{ds}X(s)$	R
Time domain integration	$\int_{-\infty}^{t} x(\tau)d\tau$	$\frac{1}{s}X(s)$	At least $R \cap (\mathrm{Re}\{s\} > 0)$
Conjugation	$x^*(t)$	$X^*(s^*)$	R

With the use of these properties we can begin to see some of the usefulness of the Laplace transform. One of the classic ways it comes in handy is in solving linear constant coefficient differential equations. In the context of continuous-time systems, those show up when a system's output is expressed as a weighted sum of derivatives of the input and output, e.g., a system with an input/output relationship similar to this:

$\frac{d}{dt}y(t) + 2y(t) = x(t)$

Now, the most general solution $y(t)$ to this equation will be the sum of a particular solution $y_p(t)$ to that equation with the homogeneous solution $y_h(t)$ (that is, the general solution to the equation $\frac{d}{dt}y(t) + 2y(t) = 0$). However, unless specifically stated otherwise, it will be universally assumed in our studies of signals and systems that the system's output $y(t)$ is 0 if there is no input (i.e., when $x(t) = 0$); that means that $y_h(t)$ is 0, so all we need to focus on is a particular solution. In order to do that, we could either take the typical differential equation solution method (using the method of undetermined coefficients, the characteristic equation, etc.), or we could use the Laplace transform.

So let's say that system has an input of $x(t) = e^{-3t}u(t)$, and we would like to find the output. This amounts to finding a solution $y(t)$ for:

$\frac{d}{dt}y(t) + 2y(t) = e^{-3t}u(t)$

To do that, start by taking the Laplace transform of both sides:

$$\begin{aligned} \{\frac{d}{dt}y(t)+2y(t)\} &= \quad \{e^{-3t}u(t)\} \\ sY(s)+2Y(s) &= \frac{1}{s+3} \\ (s+2)Y(s) &= \frac{1}{s+3} \\ Y(s) &= \frac{1}{(s+3)(s+2)} \end{aligned}$$

Partial fractions:

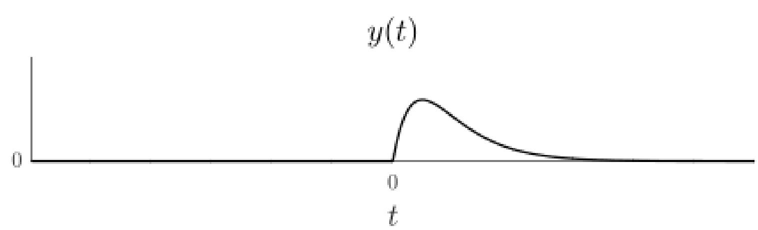
$$\begin{aligned} \frac{1}{(s+3)(s+2)} &= \frac{A}{s+3}+\frac{B}{s+2} \\ 1 &= A(s+2)+B(s+3) \\ 1 &= s(A+B)+(2A+3B) \end{aligned}$$

$$A+B=0\ ,\ 2A+3B=1 \rightarrow A=-1, B=1$$

$$Y(s) = \frac{-1}{s+3}+\frac{1}{s+2}$$

Since the system is causal, the ROC will extend to the right from its rightmost pole, i.e., $s=-2$. Using that ROC and the Laplace transform tables, we have $y(t)=-e^{-3t}u(t)+e^{-2t}u(t)$, plotted in [link].

The Laplace transform can be used to find the output when it is given in terms of a linear constant coefficient differential equation.



The Laplace transform also makes it very straightforward to find an LTI system's transfer function $H(s)$, and then by inverting it, the impulse response $h(t)$. Because convolution in time corresponds to multiplication in the Laplace domain, that means $y(t)=h(t)*x(y) \leftrightarrow Y(s)=H(s)X(s)$, and thus, $H(s)=\frac{Y(s)}{X(s)}$. Let's find the

transfer function from our example above (plotted in [link]):

$$\frac{d}{dt}y(t) + 2y(t) = x(t)$$

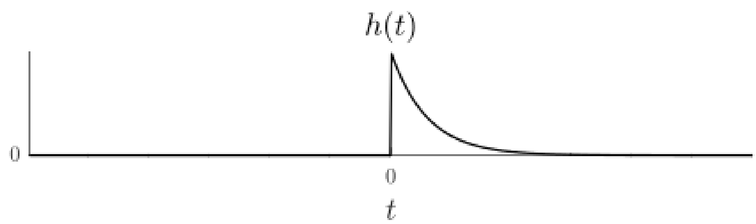
$$\{\frac{d}{dt}y(t) + 2y(t)\} = \quad \{x(t)\}$$

$$sY(s) + 2Y(s) = X(s)$$

$$(s+2)Y(s) = X(s)$$

$$H(s) = \frac{Y(s)}{X(s)} = \frac{1}{s+2}\ ,\ \mathrm{Re}\{s\} > -2$$

$$h(t) = e^{-2t}u(t)$$



Circuit Analysis and System Implementation with the Laplace Transform

In our treatment of the properties of the Laplace transform we began to see how it is a useful tool, helping us to make the solving a linear constant coefficient differential equation a matter of mere algebra. But the usefulness of the Laplace transform certainly does not stop there.

Dealing as it does with continuous-time signals and systems, the Laplace transform is also helpful as we consider the physical realization of those systems. With the Laplace transform we can easily perform circuit analysis with time varying signals, and it also gives us a straightforward way to implement systems with analog components.

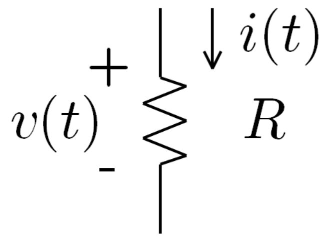
Circuit Analysis

In circuits with voltage sources, current sources, and resistors, analysis involves using Kirchoff's circuit laws and algebraically solving a set of equations. Things get trickier, however, when capacitors and inductors are introduced into the circuits. The current through a capacitor and the voltage across an inductor are both expressed in terms of differentials. Analyzing such circuits would therefore involve solving differential equations.

But as we already know, the Laplace transform can make solving differential equations easier. As a result, it also makes circuit analysis easier. It does that by generalizing the notion of circuit element resistance to be something called **impedance**.

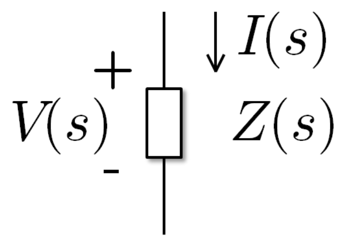
According to Ohm's law, the voltage drop across a circuit element equals the resistance of the element times the current passing through it: $v(t) = i(t)R$. [link] shows the circuit diagram representation.

A diagram of the voltage across (and current through) a resistor.



When the circuit is viewed in the Laplace domain, the voltage and current are the transforms of their time-domain equivalents, and resistance is now called impedance. $V(s) = I(s)Z(s)$. Sometimes, when referring to impedance, circuit elements are represented as boxes, as seen in [link].

When viewed in the Laplace domain, "resistance" is generalized to be "impedance."

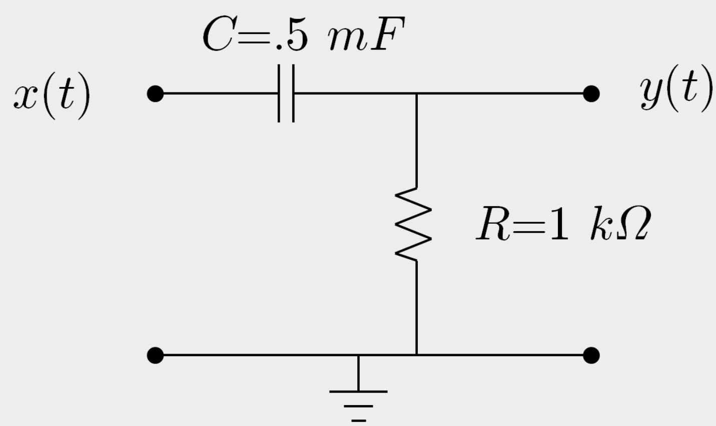


The good news of all of this is that while capacitors and conductors do not have a defined "resistance" like a resistor does, they do have readily defined impedances:

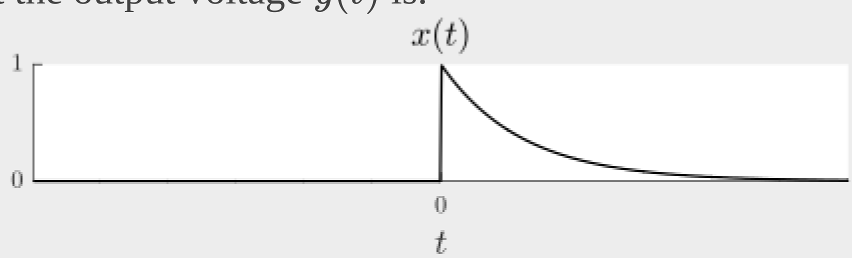
- Resistor: R
- Capacitor: $\frac{1}{Cs}$
- Inductor: Ls

So to analyze circuits containing capacitors and inductors, it is useful to view them in the Laplace domain. Their impedance values are otherwise treated as their "resistance" when performing the usual KVL/KCL analysis. It is easiest to see how that all works with an example.

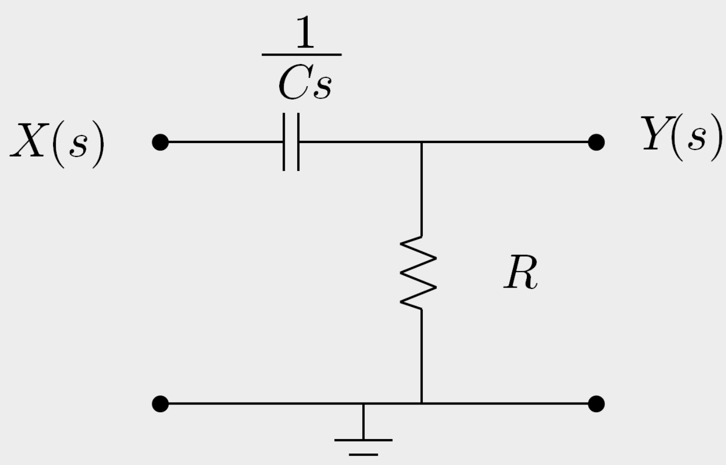
Example:
Suppose the circuit of [link] is initially at rest (no current going through it, and no stored voltage across any elements).



Then a voltage source $x(t) = e^{-t}u(t)$ is applied ([link]). We would like to know what the output voltage $y(t)$ is.



Using typical circuit analysis methods, this would be a tricky problem to solve. It is made much easier, however, by looking at the circuit in the Laplace domain ([link]).



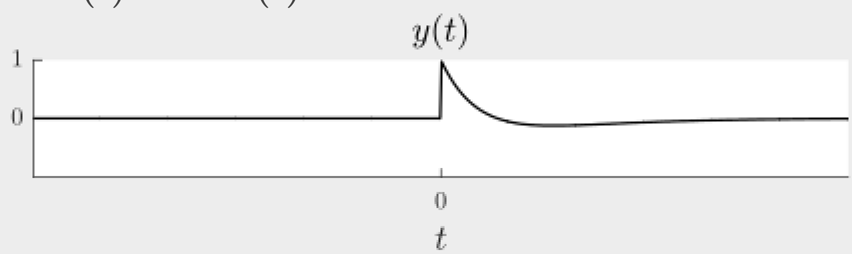
Using impedance instead of resistance, the output $Y(s)$ is found using a simple voltage divider relation. Once it is simplified enough, the inverse Laplace transform is found to determine $y(t)$ ([link]). Since the output will

be right-sided (since the system was initially at rest), the ROC will be assumed to extend rightward from the rightmost pole.

$$
\begin{aligned}
Y(s) &= X(s)\frac{R}{R+\frac{1}{Cs}} \\
&= \frac{1}{s+1}\frac{R}{R+\frac{1}{Cs}} \\
&= \frac{R}{sR+\frac{1}{C}+R+\frac{1}{Cs}} \\
&= \frac{sR}{s^2R+(\frac{1}{C}+R)s+\frac{1}{C}} \\
&= \frac{s}{s^2+(\frac{1}{RC}+1)s+\frac{1}{RC}} \\
&= \frac{s}{s^2+(\frac{1}{(1000)(.0005)}+1)s+\frac{1}{(1000)(.0005)}} \\
&= \frac{s}{s^2+(\frac{1}{.5}+1)s+\frac{1}{.5}} \\
&= \frac{s}{s^2+3s+2} \\
&= \frac{s}{(s+2)(s+1)} = \frac{A}{s+2}+\frac{B}{s+1} \\
&\rightarrow s = A(s+1)+B(s+2) \\
&\rightarrow A+B=1, A+2B=0 \\
&\rightarrow A=2, B=-1
\end{aligned}
$$

$$Y(s) = \frac{2}{s+2} - \frac{1}{s+1}$$

$$y(t) = 2e^{-2t}u(t) - e^{-t}u(t)$$



The Laplace transform also makes it easy to find the system's transfer function $H(s)$, and then--by taking the inverse transform--its impulse response $h(t)$. Once again, the ROC--though unstated below--is right-sided:

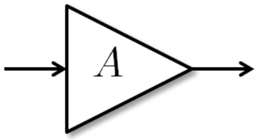
$$Y(s) = X(s)\frac{R}{R + \frac{1}{Cs}}$$

$$\begin{aligned} H(s) = \frac{Y(s)}{X(s)} &= \frac{R}{R + \frac{1}{Cs}} \\ &= \frac{Rs}{Rs + \frac{1}{C}} \\ &= \frac{s}{s + \frac{1}{RC}} \\ &= \frac{s}{s + \frac{1}{(1000)(.0005)}} \\ &= \frac{s}{s+2} \\ &= \frac{(s+2) - 2}{s+2} \\ &= 1 - \frac{2}{s+2} \\ h(t) &= \delta(t) - 2e^{-2t}u(t) \end{aligned}$$

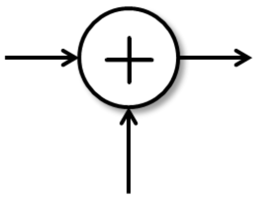
System Implementation

Using the Laplace transform, we have seen how to analyze a circuit to determine its transfer function. But suppose instead we have some desired transfer function and then want to create a corresponding circuit. We could try to make one with the usual circuit elements of resistors, capacitors, and inductors, but the more typical approach is to make a block diagram of the system using slightly more advanced analog computer components:

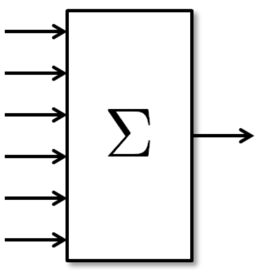
- Gain



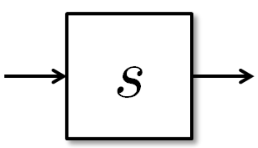
- Summer
    - Two or three inputs



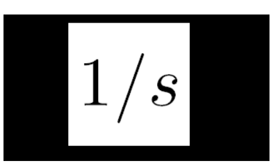
    - Many inputs



- Differentiator



- Integrator



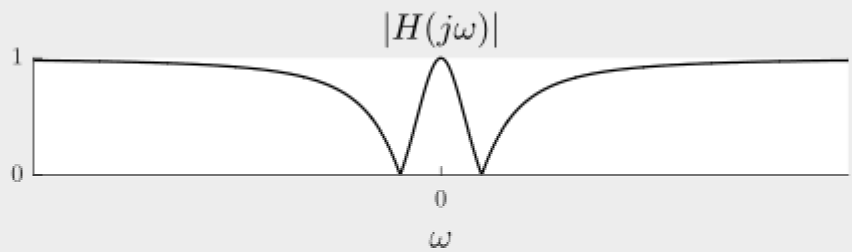
Example:

Suppose we would like to implement a system that "zeros-out" a particular frequency, but leaves the rest essentially unchanged. A system with the following transfer function will accomplish that goal, as we can see from its frequency response ([link]):

$$H(s) = \frac{s^2+1}{s^2+2s+1}$$

(recall that for causal systems, which these are assumed to be, the ROCs always extend to the right of the rightmost pole, and thus are usually left unstated)

This system eliminates a specific incoming frequency, but leaves the rest relatively unchanged.



To implement that system, we'll first create an input/output relationship in the s -domain:

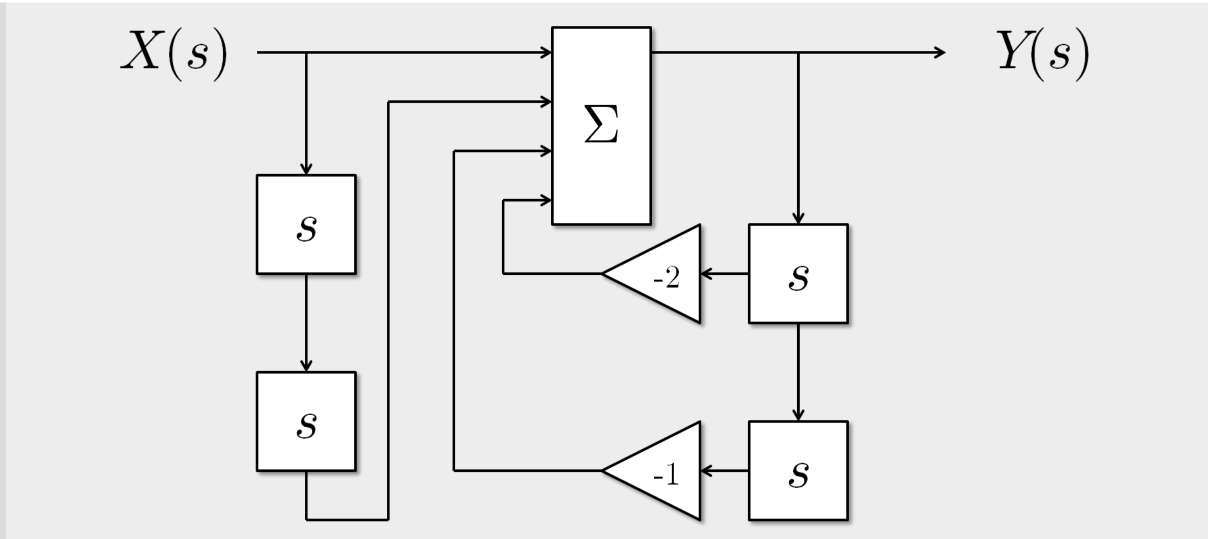
$$H(s) = \frac{Y(s)}{X(s)} = \frac{s^2+1}{s^2+2s+1}$$

$$Y(s)(s^2+2s+1) = X(s)(s^2+1)$$

$$Y(s) = X(s) + s^2X(s) - s^2Y(s) - 2sY(s)$$

From here, we can create a system diagram using gains, summers, and differentiators ([link]).

A system realization using gains, summers, and differentiators.



In the example above, we used differentiators (the "s " blocks) to implement the transfer function. In the ideal case, they work together with the gains and the summer to perfectly reproduce it. But in practice there is some amount of deviation from what the elements are supposed to do. Such deviation is only compounded by the fact that differentiators, on their own, amplify noise. As a result, integrators ("$1/s$ " blocks) are typically preferred for system implementation. The only trouble with them is that they have infinite gain on the DC component of signals, so systems that use them first eliminate the DC portion of inputs using a "coupling capacitor."

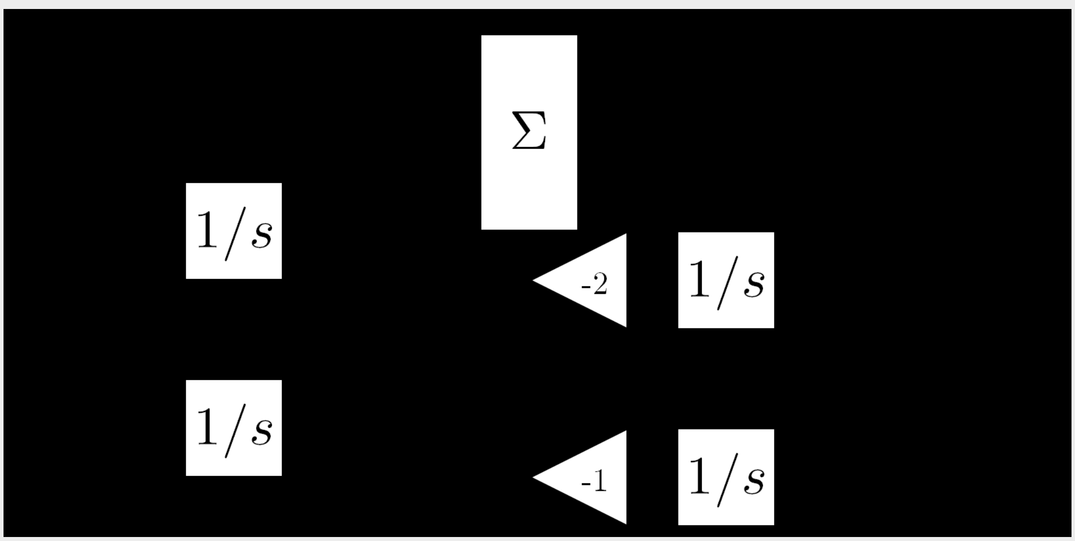
Systems are implemented with integrators in essentially the same way as with differentiators. The only difference is that $Y(s)$ is expressed using negative powers of s, rather than positive powers of s. We'll repeat the same example as before, but use integrators this time.

Example:
Using the same system as the previous example, we will implement it instead with integrators ([link]):

$$
\begin{aligned}
H(s) &= \frac{s^2+1}{s^2+2s+1} \\
&= \frac{s^2(1+s^{-2})}{s^2(1+2s^{-1}+s^{-2})} \\
&= \frac{1+s^{-2}}{1+2s^{-1}+s^{-2}}
\end{aligned}
$$

$$
\begin{aligned}
H(s) = \frac{Y(s)}{X(s)} &= \frac{1+s^{-2}}{1+2s^{-1}+s^{-2}} \\
Y(s)(1+2s^{-1}+s^{-2}) &= X(s)(1+s^{-2}) \\
Y(s) &= X(s)+s^{-2}X(s)-2s^{-1}Y(s)-s^{-2}Y(s)
\end{aligned}
$$

The same system as before, only implemented with integrators instead of differentiators.

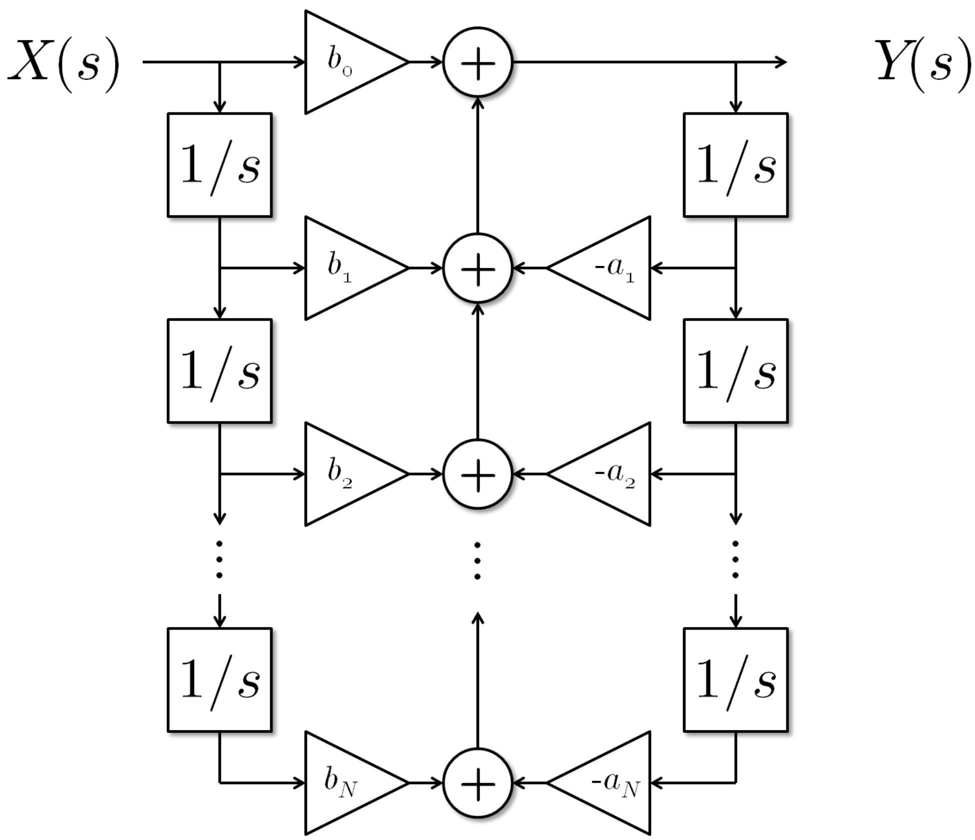


The general form for causal systems that can be implemented with gains, summers, and integrators is as follows:
$H(s) = \frac{b_0+b_1s^{-1}+b_2s^{-2}+\ldots+b_Ns^{-N}}{1+a_1s^{-1}+a_2s^{-2}+\ldots+a_Ms^{-M}}$
The Laplace domain input/output relation for the system is:

$$\begin{aligned}Y(s) =& b_0 X(s) + b_1 s^{-1} + b_2 s^{-2} X(s) + \cdots + b_N s^{-N} X(s) \\ & - a_1 s^{-1} Y(s) a_2 s^{-2} Y(s) - \cdots - a_M s^{-M} Y(s)\end{aligned}$$

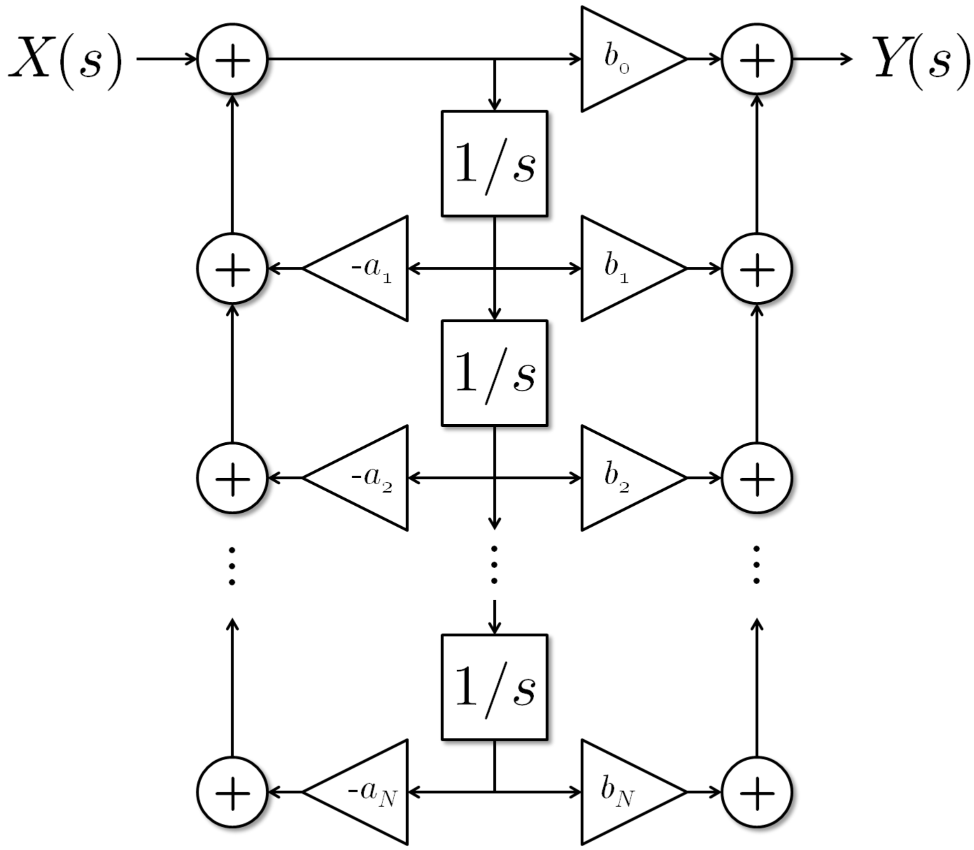
There are many ways to create a system block diagram of this system. Two standard implementations are the "direct form I" and "direct form II." The direct form I implementation has the smallest number of summers possible ([link]).

The direct form I implementation of a system has the smallest number of summers possible.



The direct form II, on the other hand, uses a minimum number of integrators. It simply flips the order of the two big parallel structures ([link]).

The direct form II implementation has the smallest number of integrators possible.



In each of the systems above it was assumed that N and M were equal. If they weren't, then the largest number would prevail in the block diagram, and some coefficients would simply be zero.

As elegant as those system implementations--or "realizations," as they are sometimes known--are, in practice they are not used, because they magnify errors of component drift (changes in component behavior). Instead, a large system is usually broken up into a series of parallel or cascaded first and second order (i.e., N is 1 or 2) systems.

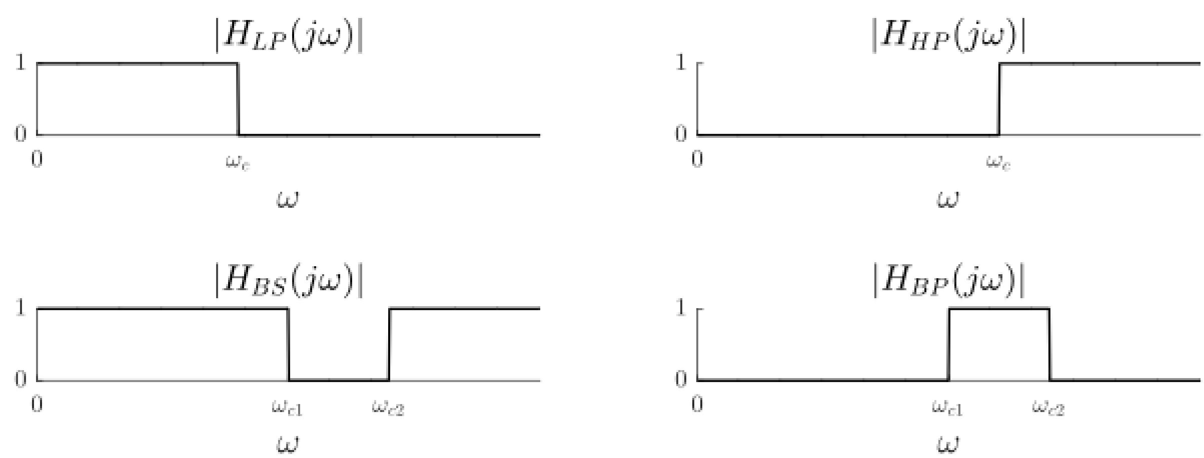
Filter Design

The above discussion on system implementation explained how a system that has a **rational** transfer function (i.e., it can be expressed as a single

fraction with a polynomial numerator and denominator) can be realized with analog computer components. But that raises a question: how are we to decide on a desired transfer function? This leads us to the topic of filter design.

When it comes to ways in which we may want to modify a signal, one of the most common is to apply a **filter**. As we would expect from the name, a filter is a system that eliminates undesired aspects of an input signal, letting the rest of it pass (hopefully otherwise unchanged). Filters are typically described in terms of what they remove and what they allow to pass in the frequency domain. A **lowpass filter** seeks to eliminate high frequencies in an input signal, and a **highpass filter** does the opposite. A **bandpass filter** lets only a certain band of frequencies through, while a **bandstop filter** removes them. If a filter is **ideal**, then its response is perfectly flat in the **passband**, and then immediately drops to zero (and remains there) at the cutoff frequency into the **stopband**([link]).

The magnitude of the frequency response of ideal lowpass, highpass, bandstop, and bandpass filters.



It is, unfortunately, not possible to create a physical realization of an ideal filter in practice. A simple examination of their frequency response illustrates why: their "boxy" shapes correspond to their impulse responses being sinc functions in the time domain. Those functions extend indefinitely

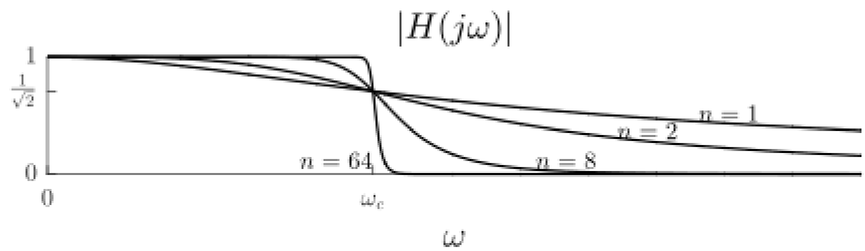
in time--past and future--so as a result it would be impossible to create a causal ideal filter (remember, the impulse responses of causal systems are zero for all $t < 0$).

But it is characteristic of engineers to find the best practical solutions available under given restraints. To that end, a number of options have been given for the problem of filtering. These options all trade-off on mutually exclusive traits:

- Flat frequency response in the passband (especially) and/or stopband
- Sharp transition from the passband to stopband
- Low filter order (i.e., small number of poles and/or zeros)

As is curiously often the case in such sets of three variables, only two can be prioritized (at the expense of the other). Looking at three of the most common analog filters will illustrate the trade-offs involved in filter design. The first is the **Butterworth filter**. This filter prizes flatness in the passband and stopbands. The frequency response of the filter is:
$|H(j\omega)| = \frac{1}{\sqrt{(1+(\omega/\omega_c)^{2n})}}$
As the filter order increases, the transition from the passband to stopband becomes ever sharper ([link]). So it is flat and can have a sharp transition, but only for a high filter order.

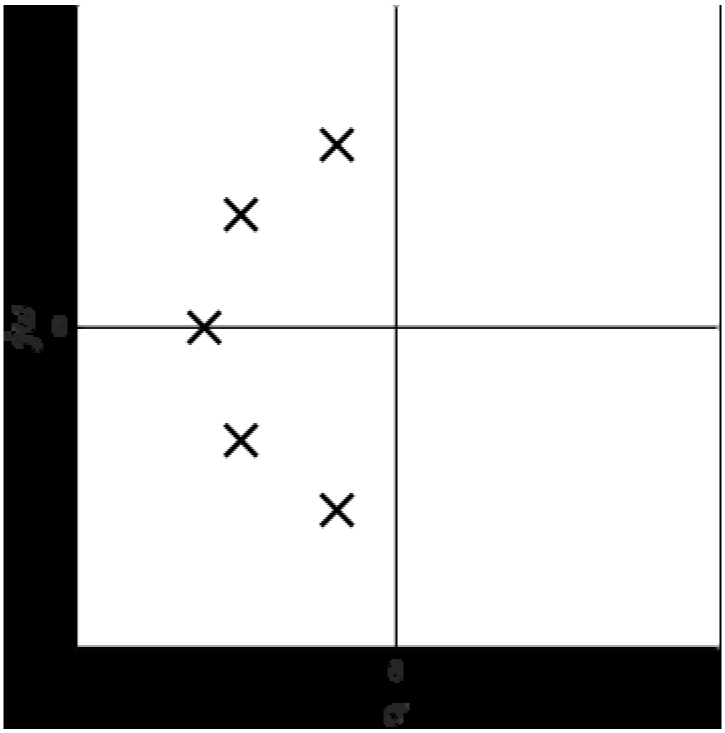
For a Butterworth filter, the passband and stopband are both flat, and the transition can be sharp if the filter order is large enough.



The implementation of a Butterworth filter is very straightforward; it is simply n poles distributed equally on a semicircle in the left-hand plane

([link]).

A Butterworth filter has n poles distributed equally on a semicircle in the left-hand plane.



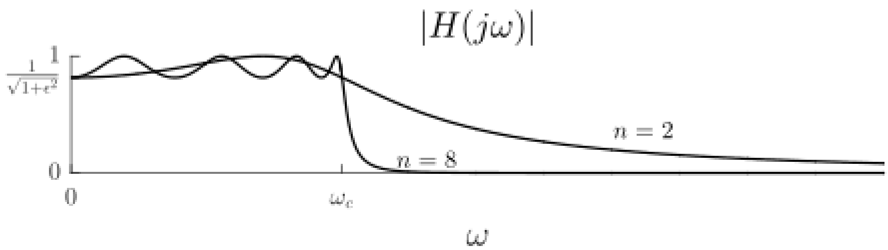
This means the transfer function is equally straightforward, a product of those poles:
If the denominator polynomial is expanded--and assuming a cutoff frequency of 1 (Laplace transform properties can be used for other cutoff values)--then we will have an input/output relationship as follows:
$Y(s) = X(s) - a_1 sY(s) - a_2 s^2 Y(s) - \cdots - s^n Y(s)$
The coefficient values can be found on a table, or by using a command in computational software. With the coefficients, an analog computer implementation can be performed.

But in the frequency response we see the tradeoff. The response is flat in the passband and stopband, but it is not possible to have both a sharp transition and low filter order. As we might expect, perhaps we can accept

less flatness in the frequency response in order to get a quicker transition. Two filters that accomplish this, in varying degrees, are the **Chebyshev** and **Elliptic** filters. The Chebyshev filter allows ripples in either the passband ("Chebyshev type I") or stopband ("Chebyshev type II"), and in return gets a sharper dropoff at the cutoff frequency. Below is the frequency response of a Chebyshev type I filter ([link]):

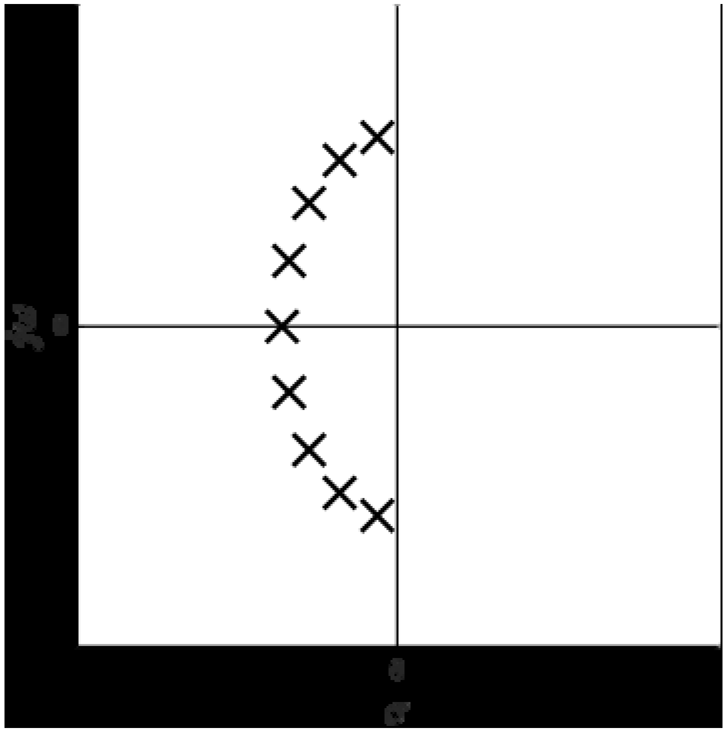
$$|H(j\omega)| = \frac{1}{\sqrt{1+\epsilon^2 C_n(\omega)^2}} \quad , \quad C_n(\omega) = \cos(n\cos^{-1}(\omega))$$

A Chebyshev filter allows ripples in either the passband or stopband. By giving up flatness in one of these bands, it can have a sharp transition with a relatively small filter order.



Note both the ripples in the passband, as well as how the transition is sharp; for a filter order of 8, it has a similar dropoff to a Butterworth filter of order 64! In terms of the transfer function, the poles for a Chebyshev type I filter are on an ellipse in the left-hand plane ([link]).

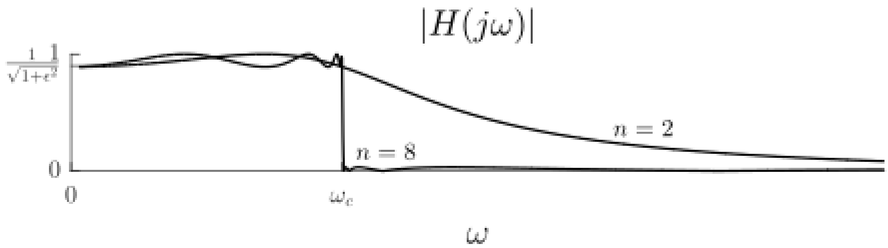
The poles for a Chebyshev type I filter are on an ellipse in the left-hand plane.



As with the Butterworth filter, the transfer function coefficients can be found in tables.

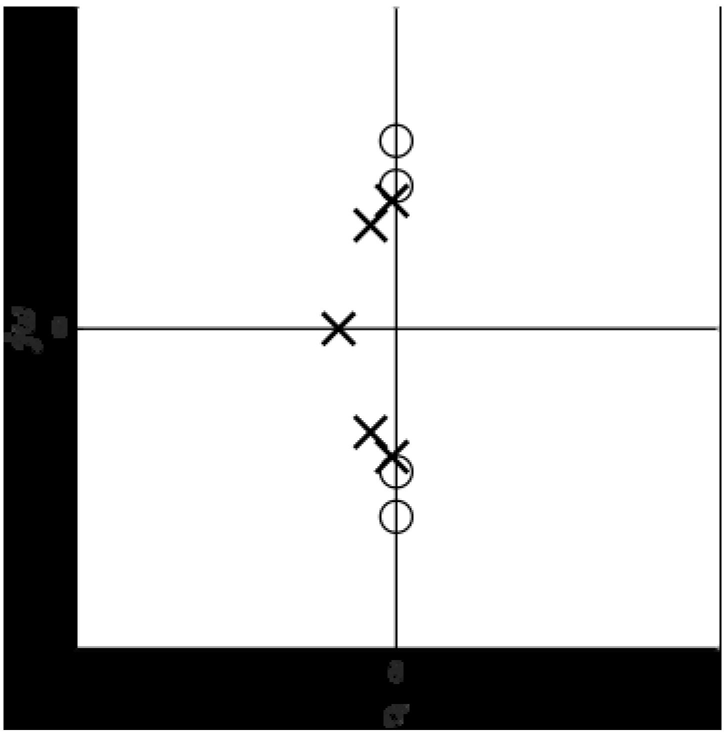
The elliptic filter takes the Chebyshev approach even further (it turns out that Chebyshev filters are actually just a special case of elliptic filters). It allows for ripples in both the passband and stopband, and therefore has the sharpest dropoff in the transition of the three filters. Its frequency response references $R_n(\omega)$, the "Chebyshev rational function" ([link]):
$|H(j\omega)| = \frac{1}{\sqrt{1+\epsilon^2 R_n(\omega)^2}}$

Because it has ripples in both the passband and stopband, an elliptic filter can have an extremely sharp transition with a very low filter order.



Here the transition from passband to stopband is nearly immediate, and with a filter order of only 8! The cost is rippling in both bands. Unlike the other filters, the elliptic filter also requires zeros in addition to poles ([link]).

Unlike Butterworth and Chebyshev filters, elliptic filters require zeros in addition to poles.

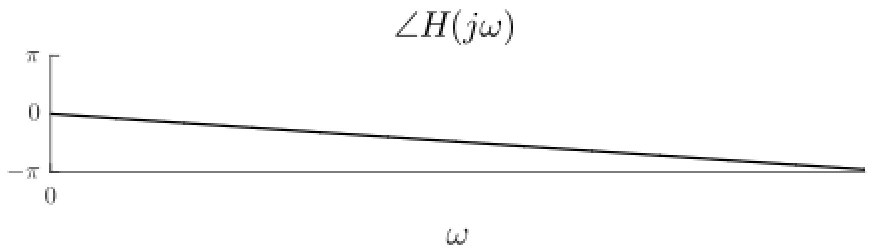


Referring to tables or computational programs is the best way to design (e.g., determine the filter order necessary to achieve certain performance, or vice versa) and implement elliptic filters.

Finally, a word is in order about the phase response of these filters. Considering only the magnitude of the filter frequency response of course does not give us a complete picture of how it modifies input signals. Not only does the filter change the magnitude of various frequency components, but also their phases.

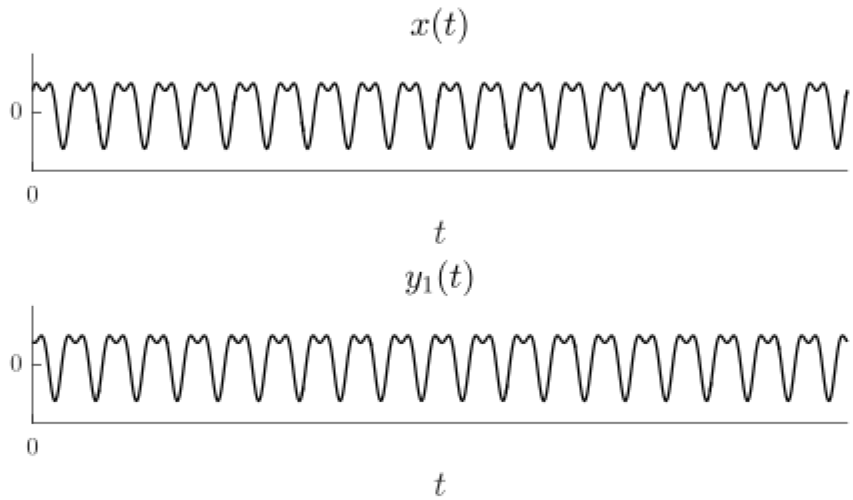
The ultimate goal for a filter is to have a "linear phase" response ([link]).

A linear phase response introduces only a simple delay into a system.



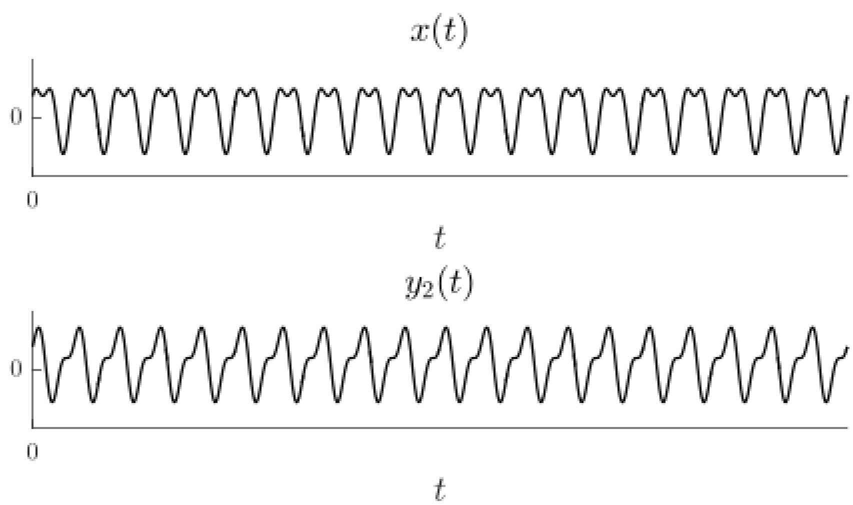
We can see why in a simple illustration. Virtually all signals are composed of a combination of different frequencies. A system with linear phase shifts all of these frequencies the same amount of distance in time (for the same shift in time, a higher frequency sinusoid shifts in phase more than a lower frequency one). The result is a delay for the whole signal. Consider a signal composed of two sinusoids ([link]). If it is put through an allpass filter (i.e., the magnitude of the frequency response is constant) with a linear phase response, the signal is simply delayed.

An input signal composed of two frequencies. A linear phase allpass system will simply delay input signals.



But suppose now the phase response is not linear. That means one of the frequencies will be shifted a different amount of time than the other, and distortion is the result ([link]).

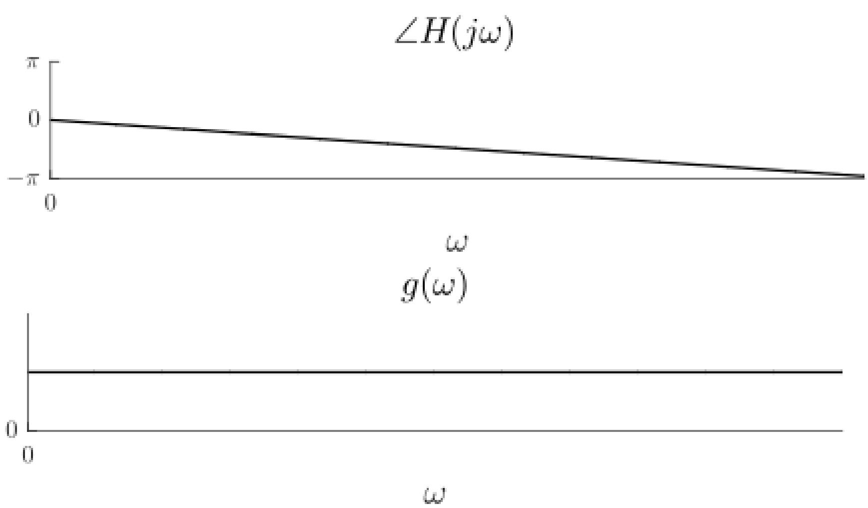
A system with a nonlinear phase response will delay different input frequencies different amounts of time.



Now, we would love to have a linear phase response; nonlinear phase response is hardly ever something that is desired. But linear phase is not something that can ever be perfectly achieved with analog filters. The best

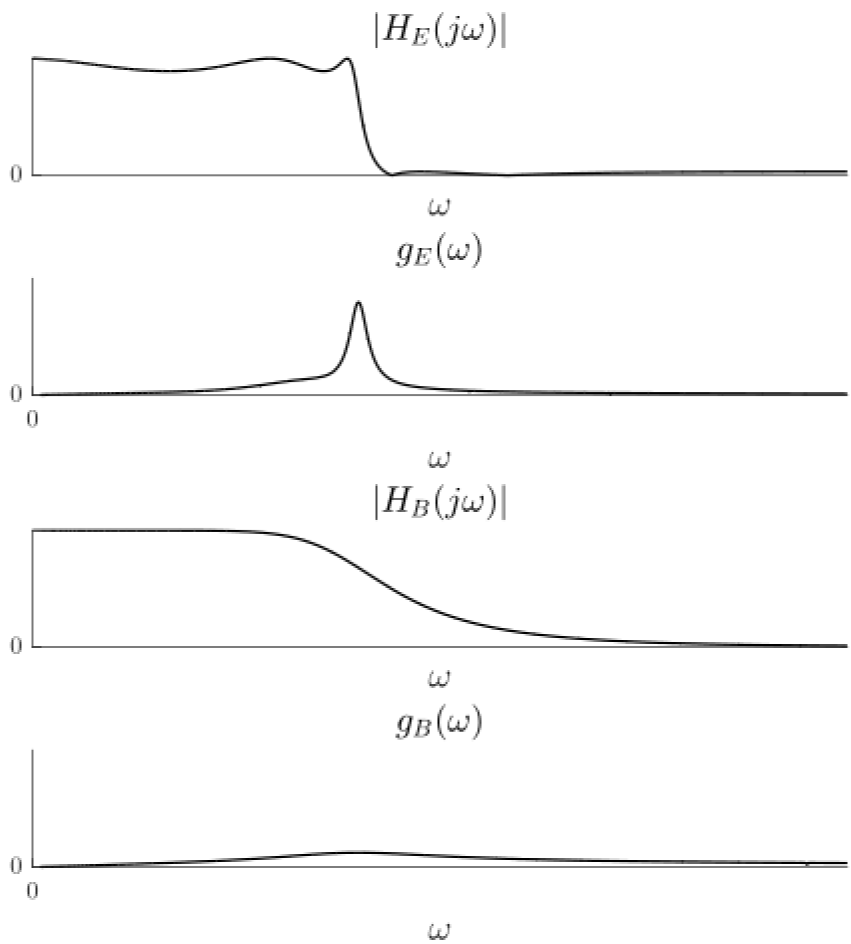
we can do is to keep them as linear as possible. To do that, an analytic tool called **group delay** is often employed.

Group delay is defined as the negative derivative of the phase of a system's frequency response:
$g(\omega) = -\frac{d}{d\omega}\angle H(j\omega)$
A linear phase response will have a group delay that is of constant value, as the first derivative of a linear function is constant ([link]).



Nonlinear phase will obviously not have constant value for group delay. Even though no continuous-time filter has a completely flat group delay, some are better than others. Compare the group delay of the following elliptic ($H_E(j\omega)$) and Butterworth ($H_B(j\omega)$) filters, each of the same filter order ([link]).

For the same filter order, the elliptic filter has a much sharper group delay than the Butterworth filter.



The Butterworth filter has a much flatter group delay, meaning it has less phase distortion. The elliptic filter (as we have already seen) has the best dropoff in magnitude between passband and stopband, but a much larger group delay, especially in the area of the cutoff frequency. So it is that phase response is yet another tradeoff variable to consider in addition to filter size, flatness, and dropoff. The particular filter you will want to choose in a given application depends on which of these characteristics you prioritize.

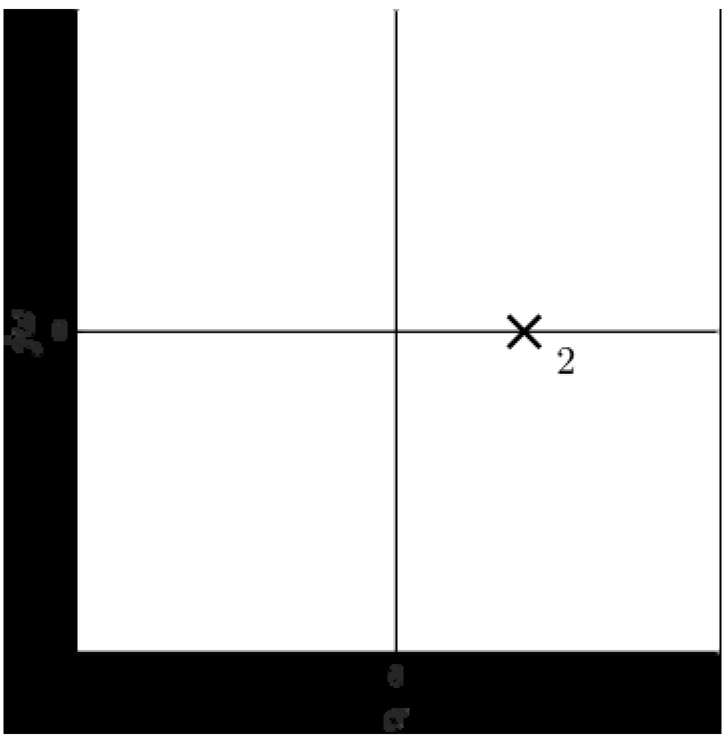
Using Feedback to Stabilize Systems

It is almost always preferable for a system to be BIBO stable (a notable exception being oscillator circuits, which start up unstable and then run on the border between stability and instability). But if a system is unstable, all hope is not lost. Using insights from the Laplace transform, we can modify unstable systems to make them stable.

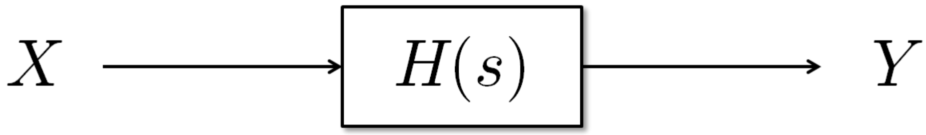
When discussing the properties of the Laplace transform, we made a connection between BIBO stability for an LTI system and that system's transfer function. Specifically, a system is BIBO stable if the transfer function's ROC includes the $j\omega$ axis. For causal systems (which are the kind we will work with when implementing systems), that means all of the system's poles must be in the left-hand plane for the system to be stable.

But suppose we are faced with a system that is not stable, such as the following very simple system:
$H(s) = \frac{1}{s-2}, \ \mathrm{Re}\{s\} > 2$
This system has a pole in the right-hand plane ([link]).

Because this causal system has a pole in the right-hand plane, it is unstable.



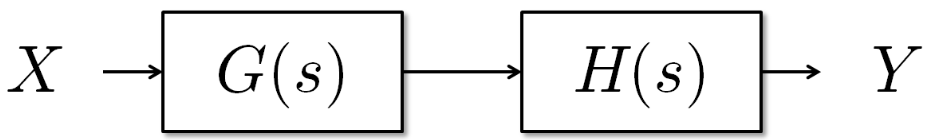
That pole being in the right-hand plane is the problem. Perhaps we could find a way to deal with it. Take a look at the system block diagram and input/output relationship in the Laplace domain ([link]).



$Y(s) = H(s)X(s)$

One way to address the unstable system H is to put another system G in front of it, and then let that system's output be the input to H ([link]):

System $G(s)$ can be placed in front of system $H(s)$ to deal with its instability.



Now the input/output relationship is:

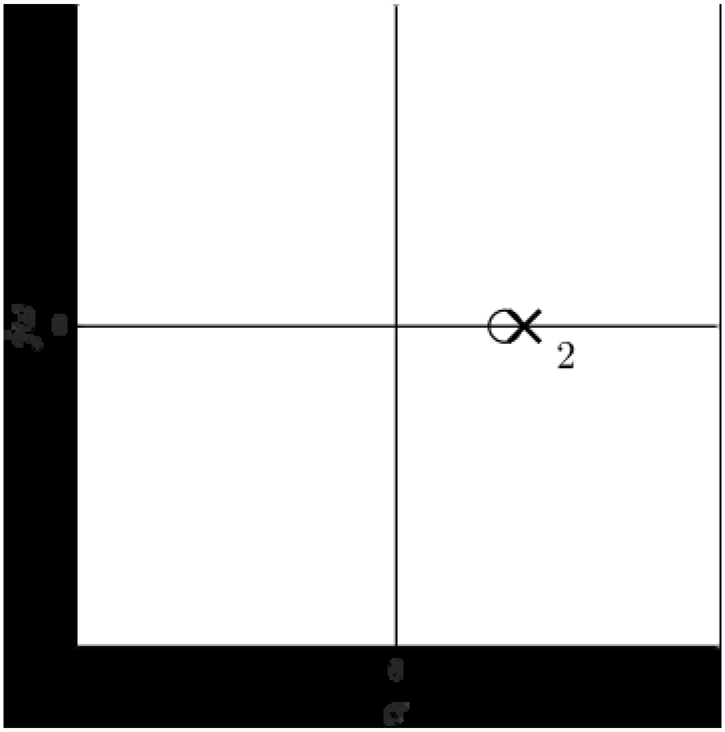
$Y(s) = G(s)H(s)X(s)$

So then the transfer function for the overall system is $G(s)H(s)$. Since the pole at $s = 2$ was the problem, we could set $G(s)$ to get rid of it by putting a zero at the same location.

$$\begin{aligned} G(s)H(s) &= G(s)\frac{1}{s-2} \\ &= (s-2)\frac{1}{s-2} \\ &= 1 \end{aligned}$$

The pole has disappeared, and the system is therefore now stable! But while this works perfectly well mathematically, there is a difficulty in implementation. If the pole and zero are not both at *precisely* the same location, then the pole still exists, and therefore the system is still unstable ([link]).

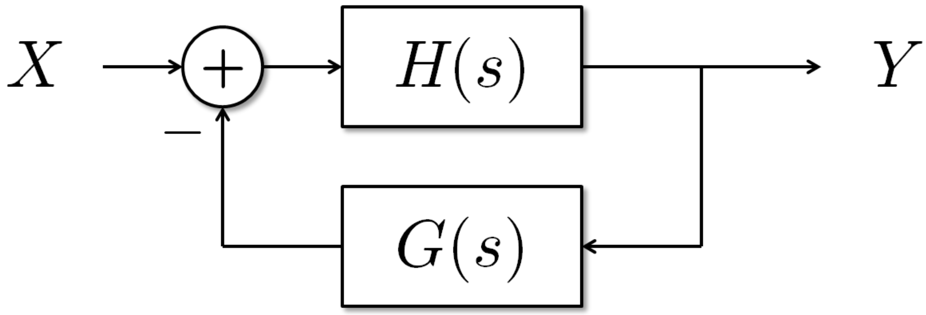
While one system's zero can theoretically cover another system's pole, any amount of component drift will uncover the pole.



So we will try another approach, called "feedback" (the previous approach is called, appropriately enough, "feedforward"). You are already familiar with the concept of feeding a system's output back into the input--it's what happens when a microphone is in front of a loudspeaker. That particular example is pretty unstable, as the sound from the speaker goes into the microphone, and then the speaker amplifies that, and so on until it becomes very loud. But there is also a way to use feedback to stabilize unstable systems.

Let's again try to use system G to deal with the pole in system H, but in a way that uses feedback ([link]).

Feedback can also be used to stabilize some system $H(s)$.



The input-output relationship for this system is now:

$$\begin{aligned} Y(s) &= H(s)\left(X(s) - G(s)Y(s)\right) \\ Y(s) &= H(s)X(s) - H(s)G(s)Y(s) \\ Y(s)(1 + H(s)G(s)) &= H(s)X(s) \\ Y(s) &= \frac{H(s)X(s)}{1 + H(s)G(s)} \end{aligned}$$

Dividing both sides of that equation by $X(s)$ gives us the overall transfer function:

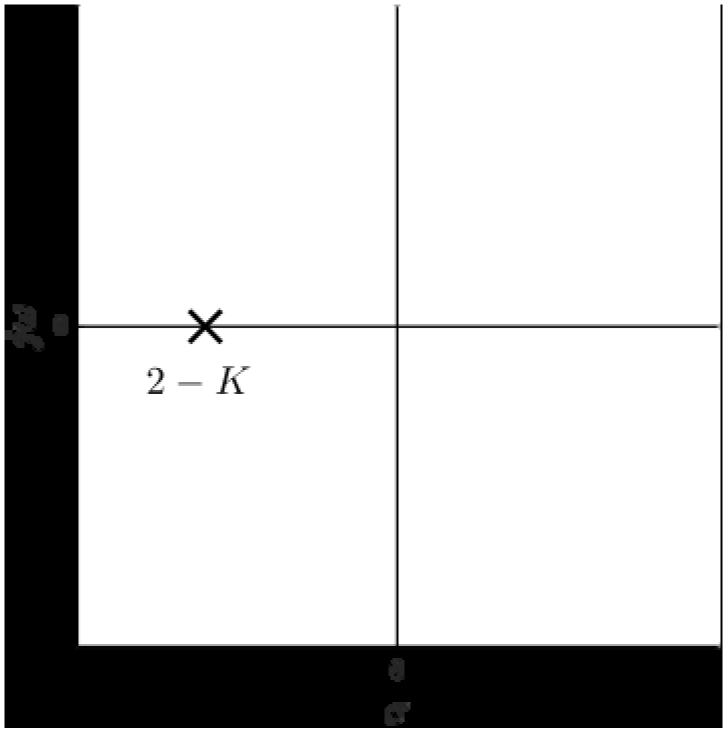
$$\frac{H(s)}{1 + H(s)G(s)}$$

So far it is not clear how this is going to help us! But let's see what happens to that pole in $S(s)$ at $s = 2$ when $G(s)$ is a very simple system that scales its input by K (i.e., $G(s) = K$):

$$\begin{aligned} \frac{H(s)}{1 + H(s)G(s)} &= \frac{\frac{1}{s-2}}{1 + \frac{1}{s-2}K} \\ &= \frac{\frac{1}{s-2}}{1 + \frac{1}{s-2}K} \frac{s-2}{s-2} \\ &= \frac{1}{s - 2 + k} \\ &= \frac{1}{s - (2 - k)} \end{aligned}$$

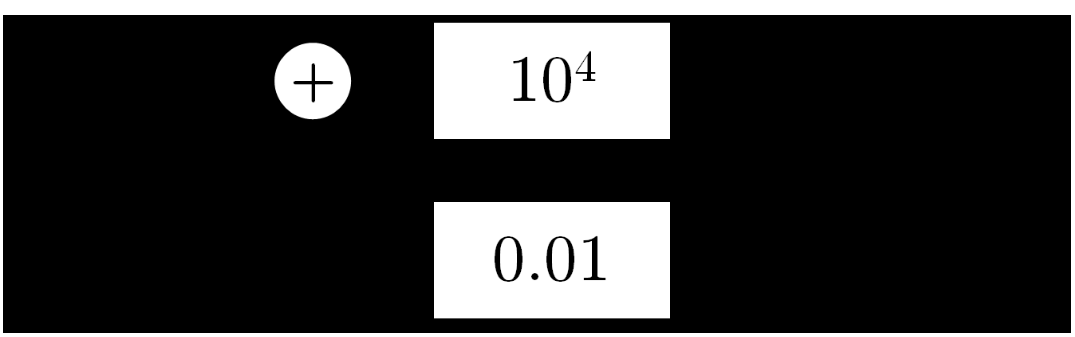
The overall system now has a pole at $s = 2 - K$. This is pretty remarkable! What the feedback accomplished was *moving* the pole at $s = 2$ to $2 - K$. So, as long as $K > 2$, the pole will be in the left hand plane, making the overall system stable ([link])!

Rather than cover a pole through a feedforward approach, feedback can be used to move a pole from the right-hand plane to the left-hand plane, stabilizing the overall system.



So feedback can stabilize a system by making a formerly unstable system to be BIBO stable. It can also stabilize a system in another sense. Consider the system in [link].

Feedback can also be used to stabilize component drift.



Let's find the transfer function of the overall system:

$$\begin{aligned}\frac{Y(s)}{X(s)} &= \frac{G(s)}{1+G(s)H(s)} \\ &= \frac{10000}{1+(10000)(0.01)} \\ &= \frac{10000}{1+100} \\ &= 99.01\end{aligned}$$

As a whole, the system is a simple gain: $Y(s) = (99.01)X(s)$. But suppose the 10^4 component drifts considerably, and doubles to be 20000. What effect do you think this will have on the overall transfer function?

$$\begin{aligned}\frac{Y(s)}{X(s)} &= \frac{G(s)}{1+G(s)H(s)} \\ &= \frac{20000}{1+(20000)(0.01)} \\ &= \frac{20000}{1+200} \\ &= 99.5\end{aligned}$$

The whole system is now a gain of 99.5. This is less than a 1% increase, even though the 10^4 component doubled. In fact, no matter how much that component increases, the overall transfer function will never be greater than 100 (and as long as the component is greater than 1000, the transfer function will be greater than 90). So the negative feedback "stabilized" the system by protecting it from component drift. It would be a different story if the .01 feedback element drifted, so that one needs to be steady. But as long as it holds to .01, it helps the overall system tolerate a huge range of drift in the 10^4 element.

These were just two simple ways feedback can be used to bring stability to a system. It has all sorts of important real-world uses. To just name applications in transportation, feedback is a key element of the controls of Segway transports, cars, airplanes, and spacecraft.